Get more characters on a line with our Spectrum utility

AN ARGUS SPECIALIST PUBLICATION

ZX COMPUTING

August September 1984

$3.95

£1.95

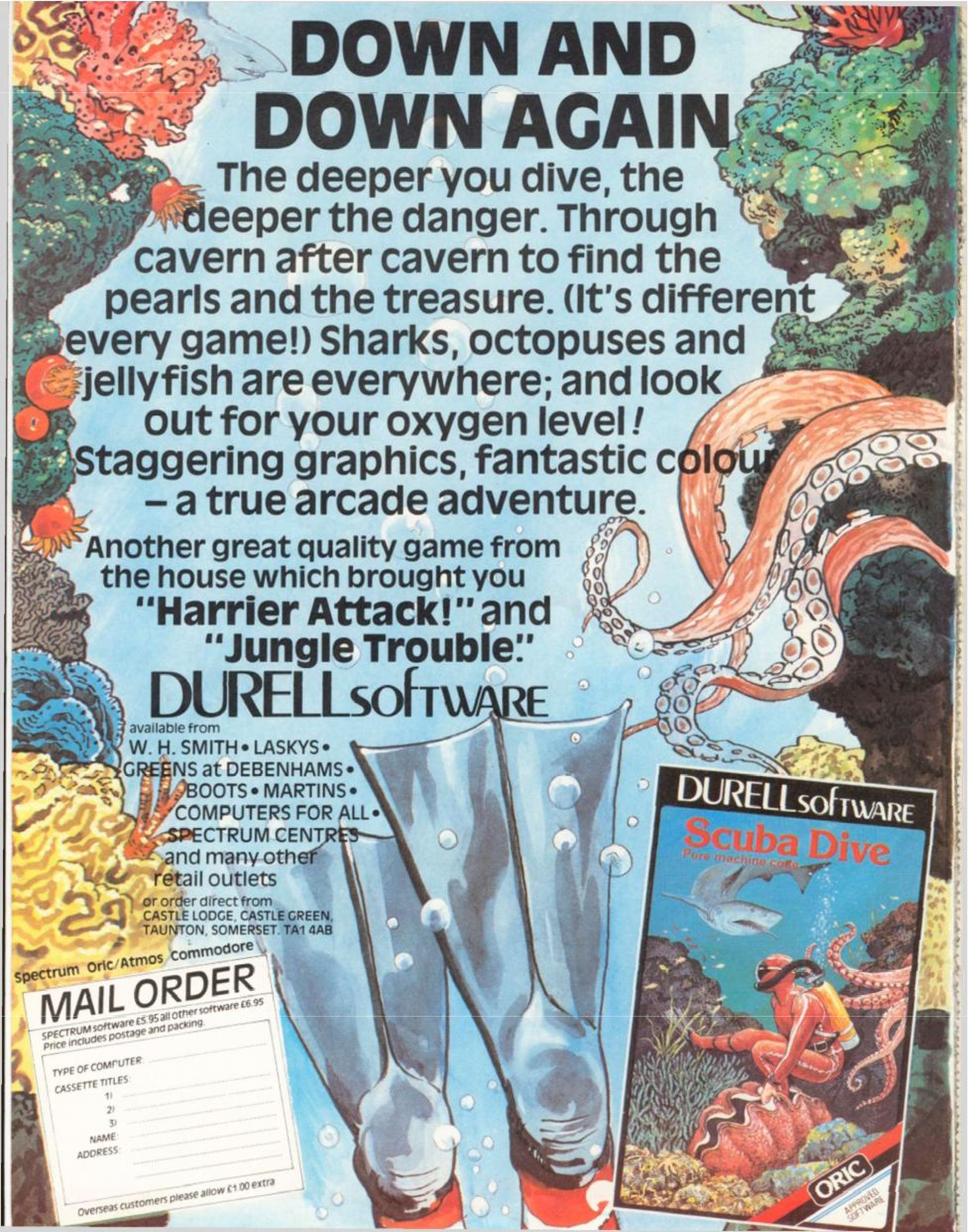
Britain's Best Buy For The Sinclair User

Blast your way through a plutonium mine

- The Streets of London – find your way with our latest ZX80 Adventure
- Raise your glasses to the Golden Chalice
- Introduce Bridge to your ZX81



Over 130 pages of information and programs for the ZX Spectrum, ZX81, and ZX80


DOWN AND DOWN AGAIN
The deeper you dive, the deeper the danger. Through cavern after cavern to find the pearls and the treasure. (It's different every game!) Sharks, octopuses and jellyfish are everywhere; and look out for your oxygen level! Staggering graphics, fantastic colour – a true arcade adventure.
Another great quality game from the house which brought you "Harrier Attack!" and "Jungle Trouble."
DURELL software
available from
W. H. SMITH • LASKYS • GREENS at DEBENHAMS • BOOTS • MARTINS • COMPUTERS FOR ALL • SPECTRUM CENTRES and many other retail outlets
or order direct from
CASTLE LODGE, CASTLE GREEN, TAUNTON, SOMERSET. TA1 4AB
Spectrum Oric/Atmos Commodore
MAIL ORDER
SPECTRUM software £5.95 all other software £6.95
Price includes postage and packing.
TYPE OF COMPUTER:
CASSETTE TITLES:
1)
2)
3)
NAME:
ADDRESS:
Overseas customers please allow £1.00 extra
DURELL software
Scuba Dive
Pure machine code
ORIC
APPROVED SOFTWARE

"I COULDN'T BELIEVE THE SPEED..... YOU'RE THE BEST BY FAR"

(Paul Singleton, Cheshire)

Thanks for all your letters, from over 40 countries. We'll continue to try to **SEND OUT YOUR PROGRAMS ON THE DAY WE GET YOUR ORDER.** And we'll send you - **FREE** - our latest catalogue of Spectrum best. We're the only company to play every single game, choose the best, leave out the rest - and then quote all the reviews, too! **ALL GAMES WORK WITH KEYBOARD CONTROL AND JOYSTIX AS SHOWN.** UK prices include VAT: export prices are the same. (This 15% surcharge helps us to get foreign orders to you very fast indeed.) Call 01-789 8546, any time, to order by Access or Visa card from any country where your own laws allow this.

The **CURRAH SPEECH SYNTHESIZER** is very popular. If you **don't** buy one, the 'Currah compatible' games work perfectly well. If you do own Currah Speech, it brings the games to life in the most amazing way!

ALL PROGRAMS REQUIRE 48K

WAR OF THE WORLDS

If you liked Jeff Wayne's LP of 'War', you'll love this animated musical-graphical adventure. You have to find six locations each on the right 'day', to rescue Carrie and save the world. Great music, beautiful scrolling animated graphics as you wander around trying to avoid the huge Fighting Machines, each with its own personality. A TRIUMPH! STIX: Protek/AGF/any cursor. **CURRAH SPEECH OK.** (CRL) **£7.95**

JACK & THE BEANSTALK

The animated graphics get better and better! As Jack, you must search the castle, without waking the giant. Watch out for the creatures and mind where you step. BUT - can you even climb the beanstalk first, without being eaten by the bugs! Very talkative, beautifully-drawn game. STIX: Kempston/Any cursor. **CURRAH SPEECH OK.** (Thor) **£5.95**

ZIG ZAG

"The 3D effect is quite fantastic" (Crash). "Excellent graphics and smooth, fast action" (S.User). Search the 3D maze for the elusive Scarabaqs, interrogate them, watch out for Hoverdroids! 6 levels. Great use of colour, graphics and speech. STIX: Kempston/Interface 2. **CURRAH SPEECH OK.** (dk'tronics) **£5.95**

THE FALL OF ROME

"A refreshing change ....a gripping game... you won't want to leave it" (HomeCompWkly) Can you reverse history as you command the Roman Empire for its last sixty years? Defeat the 11 barbarian tribes, the two Eastern Empires as you buy legions, auxilies, cavalry. Good screen prompts lead you through helpful up-dating on hi-res map of your Empire. NO STIX. (Aspi) **£6.99**

MUGSY

"A real good game wiv real A1 graphics" (PopCompWkly). It is, as the makers claim, ***"the first interactive computer comic strip".*** You run your gang of hoodlums, but all the action is in shifting comic strip frames. Even your decisions appear in speech balloons. Wonderful graphics - and it plays the Charleston, too. An original and beautiful re-creation of Chicago in the 1920's. NO STIX. (Melbourne House) **£6.95**

CODENAME MAT

"Your battle computer is as good as a flight simulation .. the astronomical details of the planets... gives the game an almost frightening realism" (CompChoice). "STATE-OF-THE-ART IN SOFTWARE" (S.User). "Amazingly realistic 3D views...93%" (Crash). The nearest you may ever get to commanding a spacefleet. STIX: Kempston/AGF/Protek/Interface 2. (Micromega) **£6.95**

HUNTER-KILLER

"An excellent program.. millions of variations" (ZX Comp). "The first real computer wargame" (Crash). CAPTAIN YOUR OWN S-CLASS SUBMARINE IN WORLD WAR 2. 18 controls for your submarine as you hunt the enemy: a control room, chart room, periscope with 360 degree view, 3D target. Quick-kill practice mode, too. Amazingly realistic! STIX: Protek. (Protek) **£7.95**

PAINTBOX

"I've never had so much fun with a utility program" (ZXComp). "A powerful graphics aid. It's possible for even a beginner to draw reasonable pictures" (S.User). "If you've been looking for a Spectrum graphics aid this seems like one of the best" (PopCompWkly). "One of the most outstanding programs I've seen this year" (YrSpec). If you want to draw, you need PAINTBOX. STIX: Kempston/any cursor. (Print&Plotter) **£7.70**

DUNGEON BUILDER

"THIS IS A MASTERPIECE... the really amazing bit is the facility to draw the pictures... outstanding" (Home CompWkly). "An excellent graphic adventure games creator which simplifies the generation of even the most elaborate adventures" (Crash). With the 15K of pure machine-code you can write your own graphic adventures. 2½ years in the making! NO STIX. (Dream) **£9.95**

OMNICALC 2

THE BEST SPECTRUM SPREADSHEET: MICRODRIVE COMPATIBLE! ***"The easiest program of all to use... the manual is by far the best" (PopCompWkly). "It stands out from the rest... it is very difficult to go wrong" (PersCompNews). "A very powerful tool" (S.User).*** The **only** Spectrum spreadsheet entirely in m/code. Up to 5000 cells. Print out OK. Built-in bar-chart routine. NO STIX. (Microsphere) **£14.95**

TASWORD 2

'IF YOU HAVE A SPECTRUM, AND WISH TO DO WORD PROCESSING ON IT, YOU COULD HARDLY DO BETTER THAN TO USE TASWORD 2" (PersCompNews). "An excellent program ... if you have been looking for a word processor, then look no further" (Crash). "Without a doubt, the best utility I have reviewed for the Spectrum" (HomeCompWkly). It is, as everyone agrees, the very best word processor for the Spectrum. Try it and see. Microdrive compatible. NO STIX. (Tasman) **£13.90**

THE FOREST

"THE WORLD'S MOST REALISTIC ADVENTURE GAME" (PersCompNews). "An extremely impressive package and highly recommended" (MicroAdventurer). "The most interesting new program for the Spectrum" (YrComp). The Forest is a computer simulation of Orienteering. Find your way through The Forest, mapping the ground. Very good graphics, illustrated 32-page manual. NO STIX (Phipps) **£9.95**

CURRAH SPEECH

Makes your Spectrum talk. Use it with talking games, and/or teach your Spectrum to say any word or sentence you like - in English. Free demo cassette and talking adventure. If you wish to use a joystick as well as Currah Speech, you will also need the **CURRAH SLOT** expandable motherboard. This lets you plug any two things into your Spectrum with/without Interface 1. **CURRAH SPEECH £29.95. CURRAH SLOT £14.95.**

ONE PROGRAM FREE!

If your order from this page comes to £25 or more, **excluding p&p,** you can choose any **one** of the programs listed below, FREE. Offer limited to supplies available - indicate second choice. If you live outside the UK, add the appropriate p&p for your free program. All programs are unused originals, which have appeared in our catalogues of Spectrum best.

MORRIS MEETS THE BIKERS (Automata) **GROUND ATTACK** (Silversoft)
ARCADE SAMPLER (Adams World) **MONSTERS IN HELL** (Softek)
MOLAR MAUL (Imagine) **SLIPPERY SID** (Silversoft) **PSST** (Ultimate)
NOWOTNIK PUZZLE (Phipps) **METEROIDS** (dk'tronics)

To: SOFTWARE SUPERMARKET, 87 Howards Lane London SW15 6NU.

If you do not want to cut this magazine, write your order out carefully on plain paper and quote this number: **ZX7**

I own a ____ K computer which is a SPECTRUM I enclose a cheque/PO made payable to Software Supermarket OR Charge my VISA/ACCESS/EUROCARD/MASTERCARD number

Signature

Please write clearly. If we can't read it, you won't get it.

Name

Address

Postcode

Phone, if any, in case of query

PROGRAM NAME		Price
		£
		£
		£
		£
		£
		£
POSTAGE	U.K. Add 55p only per order	£0.55p
AND	EUROPE: Add 55p for each program	£
PACKING	OUTSIDE EUROPE: Add £1 for each program airmail	£
	OUTSIDE EUROPE: ADD £1 TO TOTAL FOR REGISTERED MAIL — TOTAL	£

SOFTWARE SUPERMARKET

VISA/ACCESS CALL 01-789 8546 (24hrs)

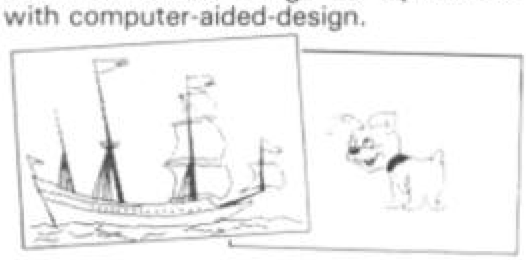
ZX COM

Editor: Ray Elder
Editorial Assistant: Jamie Clary
Group Editor: Wendy J Palmer
Sales Executive: Penny Scoular
Advertisement Manager: Barry Bingham
Divisional Advertising Manager: Beverley McNeill
Copy Controller: Ann McDermott
Managing Editor: Ron Harris
Chief Executive: T J Connell

Origination and design by MM Design & Print, Circus House, 26 Little Portland Street, London W1N 5AF.

Published by Argus Specialist Publications Ltd, 1 Golden Square, London W1R 3AB.

CONTENTS





MPUTING

ZX Computing is published bi-monthly on the fourth Friday of the month. Distributed by: Argus Press Sales & Distribution Ltd, 12-18 Paul Street, London EC2A 4JS. 01-247 8233. Printed in the UK by: Garnett Print, Rotherham and London.

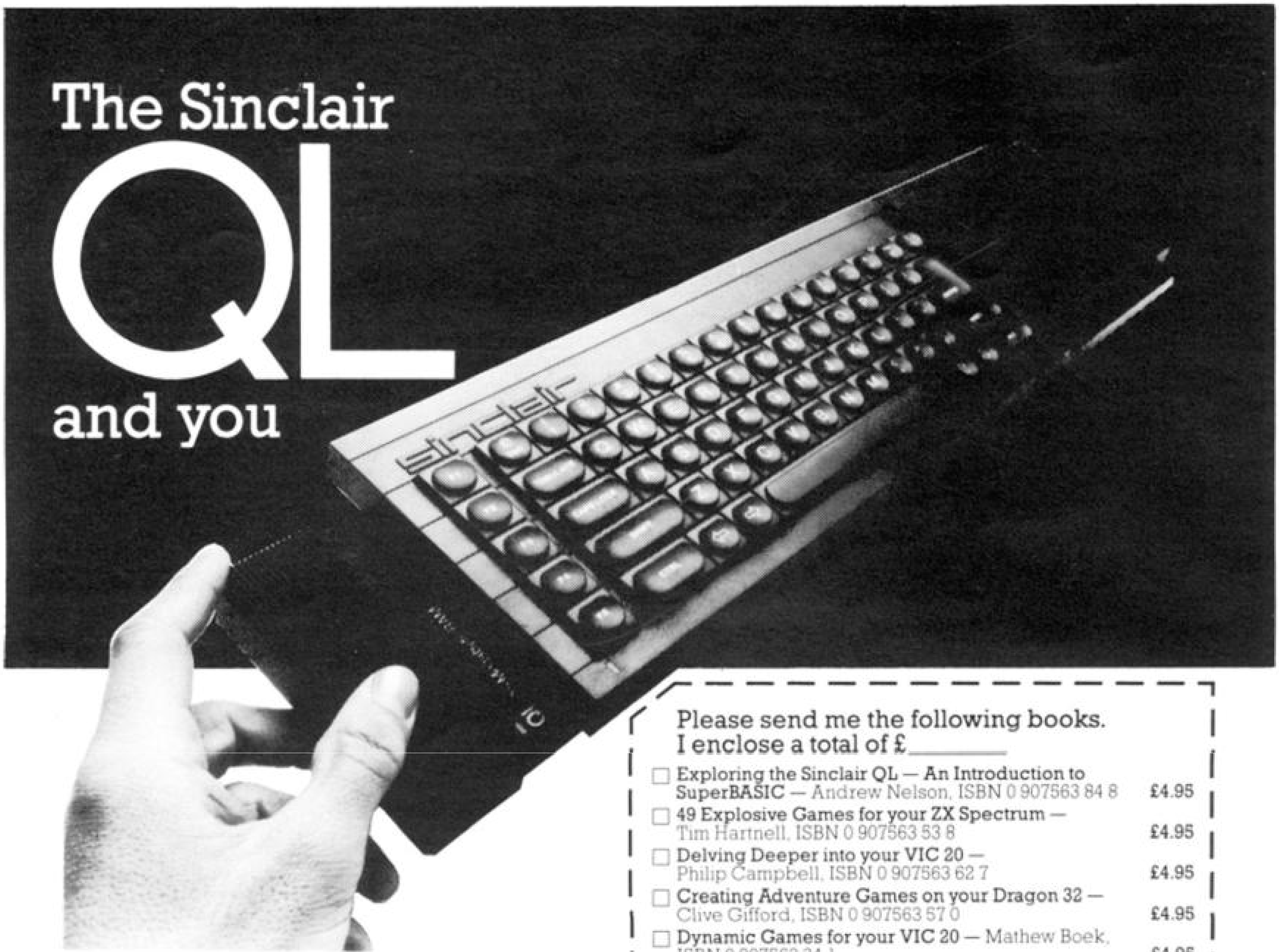


Subscription rates: UK £12.70 including postage. Airmail and other rates upon application to ZX Computing, Subscriptions, Infonet Ltd, Times House, 179 The Marlowes, Hemel Hempstead, Herts HP1 1BB (Tel: 0442 48432)

ZX Computing is constantly on the look-out for well-written articles and programs. If you think that your efforts meet our standards, please feel free to submit your work to us for consideration for publication.

All submitted material should be typed if possible; handwritten work will be considered, but please use your neatest handwriting. Any programs submitted should be listed, a cassette of your program alone will not be considered. All programs must come complete with a full explanation of the operation and, where relevant, the structure; Spectrum programs should be accompanied with a cassette of the program as well as the listing.

All submissions will be acknowledged and the copyright in such works which will pass to Argus Specialist Publications Ltd will be paid for at competitive rates. All work for consideration should be sent to the Editor at our Golden Square address.



Exploring the Sinclair QL — An Introduction to SuperBASIC
Andrew Nelson, £4.95

The QL has a rich, new programming vocabulary, and this great book gives you the chance to master the host of new words Sinclair has added. Sure, you can use the QL more or less just like a Spectrum, but if you do, you are missing the extraordinary power which lies behind the radical concepts of SuperBASIC. Command by command, statement by statement, Andrew Nelson takes you through the richness of QL SuperBASIC in *Exploring the Sinclair QL — An Introduction to SuperBASIC*. Just £4.95 and available from most book and computer shops. In case of difficulty, you can order it directly from Interface Publications, using the coupon below. (Trade only: Interface Publications are distributed exclusively in the UK and Eire by W H S Distributors. Export orders handled by Interface Publications.)

Interface Publications, Department QZX,
9-11 Kensington High Street, London W8 5NP.

Please send me the following books.
I enclose a total of £______

- ☐ Exploring the Sinclair QL — An Introduction to SuperBASIC — Andrew Nelson, ISBN 0 907563 84 8 — £4.95
- ☐ 49 Explosive Games for your ZX Spectrum — Tim Hartnell, ISBN 0 907563 53 8 — £4.95
- ☐ Delving Deeper into your VIC 20 — Philip Campbell, ISBN 0 907563 62 7 — £4.95
- ☐ Creating Adventure Games on your Dragon 32 — Clive Gifford, ISBN 0 907563 57 0 — £4.95
- ☐ Dynamic Games for your VIC 20 — Mathew Boek, ISBN 0 907563 34 1 — £4.95
- ☐ Creating Adventure Programs on your Computer — Andrew Nelson, ISBN 0 907563 36 8 — £4.95
- ☐ Putting your VIC 20 to Work — Tom Lau, ISBN 0 907563 46 5 — £4.95
- ☐ The Easy Way to Program your New Computer — Tim Hartnell, ISBN 0 907563 60 0 — £3.95
- ☐ Creating Adventures on your ZX Spectrum — Peter Shaw & James Mortleman, ISBN 0 907563 58 9 — £4.95
- ☐ Practical Uses for the Microcomputer in the Home — David Hole, ISBN 0 907563 59 7 — £4.95
- ☐ Mastering Machine Code on your Commodore 64 — Mark Greenshields, ISBN 0 907563 69 4 — £7.95
- ☐ Making the Most of your TI 99/4A — Scott Vincent, ISBN 0 907563 65 1 — £4.95
- ☐ Dynamic Games for your Electron — Neal Cavalier-Smith, ISBN 0 907563 67 8 — £4.95
- ☐ The 3-D Animated Apple — Phil Cohen, ISBN 0 907563 78 3 — £4.95
- ☐ The Art of Structured Programming — Peter Juliff, ISBN 0 907563 79 1 — £5.95
- ☐ Dynamic Games for your TRS-80 — Gail Duns & Tim Hartnell, ISBN 0 907563 63 5 — £3.95

Name ....

Address ....

....

....

INTERFACE PUBLICATIONS

We're the experts!

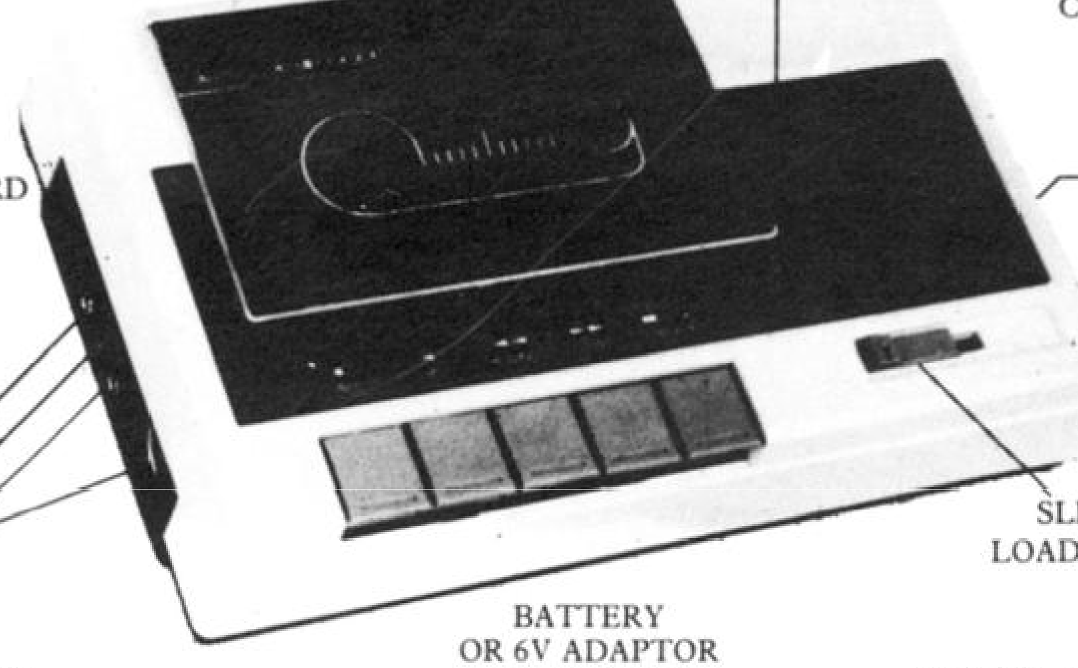
LOOK!!

THE DATA RECORDER

Meow Micros

DESIGNED FOR RELIABLE SAVING & LOADING OF COMPUTER DATA

- COMPACT SIZE *(200mm x 130mm x 50mm)*
- HIGH RECORD LEVEL
- AUTO STOP
- OUTPUT CUT ON RECORD *(leads left connected)*
- AUDIO MONITOR ON LOAD & SAVE



BATTERY OR 6V ADAPTOR
ADAPTOR SUPPLIED

SPECTRUM £24.95

ZX81 £27.95
INCLUDING INTERFACE

GUARANTEED FULLY COMPATIBLE
WITH **SPECTRUM** & **ZX81** OR YOUR MONEY BACK!!

LOOK!! SUPER SOFTWARE — NEW SPECTRUM TAPE PRICES!!

SPECTRUM 16K — PURRRFECT VALUE tapes for each computer — **ZX81 16K**

Road Race
A 3D moving road is seen through the windscreen. You must stay on the road for 45 miles. Full control of gas, gears, steering. Full cockpit display.

Soccer 6 a side game for 2 players. Rebounding ball, moving men, score set, goals, fouls. Plan view of pitch.

Golf Play your way round a 9 hole random course. Each hole is set with bunkers, out of bounds & water hazards. 3D graphics. Full set of clubs, putting, full score display.

Bar Billiards
1 or 2 Players. Computer plays 1. Red ball, break and total score, plan view of table with mushrooms.

Kludo for 2 to 6 players with rooms, weapons, suspects. Full board display, random solutions and dealing. Good fun for all the family.

Battleships
You battling the computer in this popular board game. Set your fleet, computers is random.

Stormforce
An original board game for 2 players. Each player has 6 commandos and a field gun. Capture the enemies flag. Hidden mines and action graphics.

Starbuster
You are hemmed in by a galaxy of stars, novas and super novas. Blast them to bits and move. High score chart.

Invasion Gunner
Super fast MC! 3D moving display. Lasers, shields, 2 part invaders, 3 levels. Quick reactions needed!

Master of the Rings
Avoid losing lives, Search snake pit, ravine, trees, maze, black rings, dark riders. All with traps Find 6 hidden rings. Action graphics.

SPECTRUM (£3.50 each)
£6.00 for 2 tapes

2 SUPER GAMES ON EACH TAPE!!

ZX81 (3.50 each)
£6.00 for 2 tapes

SPECTRUM 48K

NEW!!

FISHY DICK
Dick must catch the fast moving fish as they swim in his river. He can cast and strike. Mind the ducks! Beautiful graphics All machine code. The more fish you catch the harder it gets! Cast & Strike with different bait for each fish.

£3.50

LOOK!!

PROGRAMMERS WORKSHOP
- **Character Generator** block display generation of 21 UDG's + 147 extra in array storage.
- **Memory Peeker**
- **Tape Header Dissembler**
- **Attributes Calculater**
- **M.C. Sound Effects Generator** 1000's of combinations, single note, ramped twin & two-tone.
- **Number Convertor** Binary—decimal—hex up to 16 bit
- **Number Translator**
- 16 Bit To Lo & Hi order bytes in hex or decimal.
- **UDG Array Display** Block displays of stored graphics

ONLY £3.50!

ALL Tape Prices include Postage and Packing and VAT
Data Recorder ADD £1.50 Postage, Packing & Insurance
Allow 7—14 days for delivery
Send cheques or Postal Orders only
Overseas orders for software tapes only! (ADD £1.00)

Meow Micros
8 Newnham Close
BRAINTREE, Essex CM7 7PR
Tel: (0376) 22425

Welcome

Too much, less, more, great, rubbish, not enough, why don't you, and so on and so forth. . .

To paraphrase: You can please all readers some of the time, and some readers all of the time but never all readers all of the time!

What am I waffling on about this time?

Regular readers will recall that I published a questionnaire way back in the FEB/MAR issue. At long last I have managed to sit down and correlate all the information and, as promised, here is a breakdown of the results.

Somewhere on this page should be a box with the figures set out. I wrote a simple program top calculate these figures (which should be lurking around in this section somewhere for those interested), and myself and two helpers whose aid was enlisted by threatening them with money, sat down and entered the info. The experts say that statistics can be used to prove anything, I will explain how I interpret the figures.



Assessment

Although 1496 is only a small proportion of the total readers — thank goodness there were no more, these results still wouldn't be ready! — it is enough to take as a proportional representation.

If you look at the age range you will see that the replies are from an evenly spread group. Each of the sub-headings in the next two sets, User and Reader, are to be taken as individual percentages and do not relate to each other as several combinations of each sub-head are possible. ie. The computer could be used for any combination of games, programming and 'other' in equal amounts, so many readers ticked two or even all three boxes.

As far as the proportion of readership is concerned I think that the distribution is a fair reflection for the purpose of evaluation.

The sharper one among you will have noticed that the totals do not usually add up to 100%, this is due to the fact that I used INT calculations which does not round up to the nearest whole number. However for my purposes the results are significant regardless.

Notably, a significant number of readers feel that too many games programs are published, and too little space is allowed for utilities, hint/tips and machine code.

Generally the majority of readers seem to feel that the rest is about right.

Action

These trends were noticed as the forms were being gathered together and I have already tried to include more utilities in each issue, I will continue to try to keep this balanced.

Nearly all requests for more machine code were accompanied by a plea for simpler progrmas or a beginners' series, at the same time a great deal of praise for the standard of Toni Bakers programs was received from the more advance programmers. I am working on a way to get the best of both worlds.

The highest area where you feel that more space should be given is for Hints and Tips. I shall try out a couple of pages in the next issue (if possible) and see how it works out.

Answers

Although I just haven't the time to answer all the comments you made I have read them all and considered each one. Some brief replies I'd like to make are to:

J Booker, Yes and Yes.

A J Harper, hardware construction articles herein.

J Bourne, Book and Education reviews included.

W Dewhurst, sorry about your ZX81 failing after 3 months.

R W J Payne, good idea, I'll try and find someone with the knowledge.

Mike Jepson and Arie Baars, sorry folks, we haven't any copies, may I recommend Toni Baker's book "Mastering Machine Code on Your ZX Spectrum" published by Interface.

Roy Ivan Powis, so do I.

M F Veevers, agreed, how about writing me one!

Mrs. V Jones, 17 Roding Drive, Kelvedon Hatch, Brentwood, Essex CM15 0XA requests "older women" to get in touch with a view to improving her ability.

S C Arnold, interesting regular reading habits.

S Franklin, F J Gardiner and M Samuels don't like new listings, sorry folks, you're in the minority but entitled to your opinion.

J Piacentini, yes 16K Spectrum progs will run on a 48K.

Miss K D Yarwood, a page for females would be sexist!, all our pages are bisexual (no reflection on the editor)

G Williams, agreed, 1 & I too similar, the ∧ is the arrow on key H.

B R T Owen, well spotted.

G Paraskevaidis, 1K Corral fans would lynch us, note lack of machine specs.

Pedro Inglesias, sympathies on being charged 725 pesetos (£3.35) per issue, to A Claessen in the Netherlands who pays HFL16 (£4.00) and S. Barry in Dubai who parts with £9.95!

Rudy Meijering, thanks.

E R Shepherd, yes you are!

Generally

Although I'm frustrated at the failure to trap all errors and disappointed that some mistakes still get through, I will continue to fight against the bugs with the goal of producing a "perfect" issue.



Finally

Thanks to all of you who bothered to send in the questionnaire and I hope that you will feel that some improvement has resulted.

USA Users

At the last moment, just before going to press, I made two phone calls to the United States, one to Mary Rhineman of Sinclair Research and one to Timex.

Although Timex has withdrawn from the market and Sinclair does not intend to sell the Spectrum in the USA. — the QL will be on sale though — Timex is negotiating for a company to market a cartridge which will make the Spectrum software compatible with the TS2068.

This will open up a whole new range of wares for the machine. Many requests for help with hardware compatibility have also been received and a couple of readers sent details of the TS2068 from which, initially, there seems little hope.

Nevertheless I hope to bring you some good news in the very near future once I have consulted with experts in this field. Watch this space!

and so ONCE MORE INTO THE BREACH. . . . .



Results as percentages of 1496

Age:	*Under 21* 38%	*21-35* 32%	*Over 35* 29%
Use:	*Games* 67%	*Programming* 77%	*Other* 22%
Reader:	*Regular* 67%	*New* 32%	*occas'n'l* 6%
Space:	*Too much*	*OK*	*Too little*
Games	38%	50%	11%
Util's	2%	39%	58%
Reviews	9%	70%	20%
News	6%	75%	18%
Articles	3%	74%	21%
General	5%	80%	14%
Hints	1%	28%	70%
M/Code	8%	34%	57%

Competition Blunder?

Dear ZX Computing
In the Jan/Feb edition of "ZX Computing" I noticed that in the competition, Ray Elder made a mistake and had written "David Knight" and it was meant to be "Michael Knight".
Yours,
Mandy Stenhouse, Simon Baizley
Minehead
Somerset.

Prevention

Mr. H T Walker of Guildford and Neil Burton of Strathclyde both sent in this tip which could help you out of an awkward situation.

Some programs prevent you from stopping it at the INPUT stage by using the LINE function — try this:

```
10 INPUT LINE a$: PRINT AT 0,0 ;a$: GOTO 10
```

You will find that is nearly impossible to break this program, however by pressing CAPS SHIFT and key 6 at the same time the program will stop with a "H STOP in INPUT" report. Very useful if you get into a closed INPUT loop. . .

Spectrum Set-Up

Dear Mr Elder,
Having just read through the latest edition of ZX Computing I thought you might like to hear of my experiences regarding the use of the Spectrum for the word processing and data retrieval.

My present set-up consists of a 48K Spectrum driving a Shinwa CP80 printer via the Kempston centronics interface. In addition I have microdrive and interface 1. The computer and Interface 1 are mounted inside a DKtronics keyboard.

The software I use consists of Tasword and Campbells systems MASTERFILE.

Firstlty regarding the keyboard. I have had to cut away most of the rear panel of the case to accomodate the computer + Interface 1. The benefit of this setup is that the expansion port of interface 1 now projects from the back of the keyboard eliminating the connection problems previously experienced.

Secondly, concerning the Kempston Interface. The current model only allows the screen copy to occupy about half of the available printer width. This is not only a waste of paper but also leads to undersize printouts of programs such as Masterfile.

I recently noted that Kempston have launched a new model E centronics interface which allows double width copies. (A feature apparently offered by the Tasman interface). On contacting Kempston to enquire if it was possible to upgrade the ealier interface to give this feature I was told that it was possible by writing a new

```
  10 LET count=0: DIM a(11,3)
  19 REM Start
  20 LET count=count+1: CLS
  30 PRINT "Is the age:"''"1 les
s than 21"''"2 between 21 and 35
"''"3 over 35"
  40 LET n=1: GO SUB 9000
  50 FOR j=2 TO 3: FOR i=1 TO 3:
 CLS : IF j=3 THEN  PRINT "Reade
r type?"''("Regular" AND i=1)+("
New" AND i=2)+("occasional" AND
i=3)
  55 IF j=2 THEN  PRINT "Use:"''
("Games" AND i=1)+("Programming"
 AND i=2)+("Other" AND i=3)
  60 INPUT "Enter Y or N "; LINE
 a$: IF a$<>"y" AND a$<>"n" THEN
  GO TO 60
  70 IF a$="y" THEN  LET a(j,i)=
a(j,i)+1
  80 NEXT i: NEXT j
  90 RESTORE : FOR n=4 TO 11: CL
S : READ s$: PRINT "Allocation o
f Space:"''s$''"1 Too much"''"2
Just right"''"3 Too little"
 100 GO SUB 9000: NEXT n
 110 INPUT "Any More ? ";a$: IF
a$(1)="y" THEN  GO TO 20
 120 IF a$(1)<>"n" THEN  GO TO 1
10
 200 CLS : PRINT "Results as per
centages of ";count
 210 PRINT '"Age : Under 21  21-
35  Over 35"
 220 PRINT TAB 9;INT ((a(1,1)/co
unt)*100);"%";TAB 17;INT ((a(1,2
)/count)*100);"%";TAB 25;INT ((a
(1,3)/count)*100);"%"
 250 PRINT '"Use : Games   Progr
amming  Other"
 260 PRINT TAB 7;INT ((a(2,1)/co
unt)*100);"%";TAB 18;INT ((a(2,2
)/count)*100);"%";TAB 28;INT ((a
(2,3)/count)*100);"%"
 270 PRINT '"Reader : Regular  N
ew  occas'n'l"
 280 PRINT TAB 11;INT ((a(3,1)/c
ount)*100);"%";TAB 18;INT ((a(3,
2)/count)*100);"%";TAB 26;INT ((
a(3,3)/count)*100);"%"
 290 PRINT '"Space: Too much  OK
  Too little"''
 300 RESTORE : FOR i=4 TO 11
 310 READ s$: PRINT s$;TAB 9;INT
 ((a(i,1)/count)*100);"%";TAB 17
;INT ((a(i,2)/count)*100);"%";TA
B 25;INT ((a(i,3)/count)*100);"%
"
 320 NEXT i
8999 STOP
9000 INPUT "Press 1,2 or 3: "; L
INE a$
9010 IF a$="" OR LEN a$>1 OR a$<
"1" OR a$>"3" THEN  GO TO 9000
9020 LET a(n,VAL a$)=a(n,VAL a$)
+1
9030 RETURN
9999 DATA "Games","Util's","Revi
ews","News","Articles","General"
,"Hints","M/Code"
```

Editors number-crunching program.

Machine code routine. Kempston however have no plans to produce such a routine and are concentrating on selling their new Interface, a somewhat annoying attitude for those of us who have the old model!!

A possible solution to this problem is a new package soon to become available from Microdot software (see ad. page 98 *ZX Computing* Apr/May 84). I phoned Microdot who were very helpful and informed me that their "Phantom Utility" program would more than rectify the shortcomings of the Kempston interface. The utility is expected to be available around the end of April and they have promised to let me know when it becomes available.

I will send you my findings on the Microdot package if you are interested. Keep up the good work.

Yours sincerely
G C Meikle

Glorified Catalogues

Dear Sir,
I was recently on holiday in the U.K. when I happened to pick up the *ZX Computing* magazine. What a relief to find a publication that actually serves the public. Over here most magazines have one very clear purpose — to sell, sell, sell personal computers. Clearly the very idea giving away free information, programming tips and to actually print out games programmes, is beyond most editors concept of good business practice. They miss the fact that their readers soon give up after 5 or 6 issues recognising the magazines for what they are — glorified sales catalogues. So many thanks to ZX and many thanks to the readers that help make your magazine what it is. I only hope the UK readers know and appreciate what they have.

Sincerely
David A. Hinson
Bethesda, U.S.A.

P.S.
1/ Is there anyone who could help me convert your Spectrum programmes to suit the Timex Sinclair 2068?
2/ I take it that the $ signs shown on the front of ZX refers to Australian sales price. Can it be bought in the States? I get my copy regularly but other people in Sinclair Users Clubs would like to get hold of it too.
3/ The Timex Sinclair is being discounted at present for less than $100. Normally between $150 and $200. The TS 2040 printer (an Alphacom 32) can also be purchased for as low as $50 (Normally $79). So anyone planning a trip to the States could benefit substantially if they manage to find one at these prices!

Long Live The ZX81

Dear ZX Computing
Having just read your inaugural editorial and sufficiently recovered from the humour (only joking) I thought I might add to your comments on the ZX81 (16K). I agree entirely that it has plenty of life in it yet, and even in years to come when computers have attained Artificial Intelligence status I'm sure it will be regarded in retrospect as an all-time classic, with a certain charm and beauty all its own. It's a pity that some of the more fiscally motivated producers and particularly the purveyors of software have abandoned it so soon, but three cheers for the continuing support from *ZX Computing*.

Despite (or is it because of) the ZX81's obvious limitiations I enjoy trying to squeeze as much out of it (or into it!) as possible, and the occasional victory in this "battle" can be very satisfying. For those who may not understand this it could be loosely compared to the amateur radio enthusiast who delights in achieving contact halfway round the world using what may be theoretically an insufficient signal strength, when of course he could have 'phoned.

After experiencing the Basics (capital B) of such as the Texas TI99, the Dragon and the BBC I am even more grateful for nice uncluttered Sinclair Basic, especially its editing and its string slicing. The latter are not the sort of things that impress the average 'spec-sheet-enthusiast', but they are in fact just as powerful, yet simpler, and clearer for thinking with than the messy systems used elsewhere. Virtually unrestricted string length, line length, and total freedom for substitution of expressions are also typical of the many logical and instinctively 'natural' features of Sinclair Basic that I am amazed to find missing in many other (supposedly superior) dialects. Functions that are not included 'ready made' can nearly always be synthesised when needed, with the well known ingredients 'existing commands plus a little mental effort', although some machine code seasoning may be advisable to recover speed!

Now before anybody dismisses my liking for the humble ZX81 as being just stick-in-the-mud let me say that I **do** like hi-res and colour, I am aware of the power and speed of other machines, and I appreciate all the latest advances as much as anyone. However, an intelligent and (above all) **honest** mind can **also** find and enjoy higher aesthetic qualities in places where less perceptive types would need to be swamped with unsubtle excesses to notice anything at all!

Yours sincerely,
J. Dave Rogers

Helpful Hints

Dear Sir,
I send to your magazine a typewriten page with some hints of the Spectrum's ROM, which will be certainly very useful for all your readers.
Thanks
Moreau Stejhane

The ZX Spectrum's ROM contains some routines which can be very useful in your programs.

- Free memory (16-48K): PRINT 65536 — USR 7962 ; "bytes".

- Scrolls :

RANDOMIZE USR 3280 : scrolls whole page up one line.
RANDOMIZE USR 3330 : scrolls whole page to first line.
RANDOMIZE USR 3582 : scrolls whole page up one line.(BORDER!)
RANDOMIZE USR 3583 : scrolls half bottom page up one line. (BORDER!)
RANDOMIZE USR 3652 : clears half bottom page. (BORDER!)
POKE 23292,X affects scroll. (see Sinclair manual, Chap. 15 Page 106)

- Change cursor : (upper/lower case)

This depends on the 4° bit of FLAGS 2 at address 23658.
POKE 23658,0 : gives lower case, 4° bit=0
POKE 23658,8 : gives upper case, 4° bit=1
RANDOMIZE USR 4317: changes caps to lower case, and vice-versa.
POKE 23617,236 : to have a ? during inputs.

- Protections :

POKE 23659,0 :	disabling the BREAK key, but the program can't produce reports! (no stop, no bug. . .)
POKE 23606,X :	deteriorates character set. (X=76/92/108/111/164. . .) X=0 restores normal character set. Try also with 23607
POKE 23756,0 :	the first line will become line ZERO.

- Another method for your programs:

```
1 LET A=PEEK 23637+256XPEEK 23638:POKE A,O:POKE A+1,0
2 REM !!!!!!!!!!!!!!! LINE ZERO !!!!!!!!!!!!!!!
```

- A new kind of DRAW can be obtained with the X co-ordinate of last point plotted at address 23677 and the Y co-ordinate of last point plotted at address 23678.

- Save memory :

Change 0 for NOT PI ; 1 for SGN PI ; 3 for INT PI.
Use VAL "number" for the other numbers.
1000000 becomes VAL "1000000" or better VAL "1e6"
Change PRINT AT X,0;"32 spaces"
to PRINT AT X,0;,,

- New :

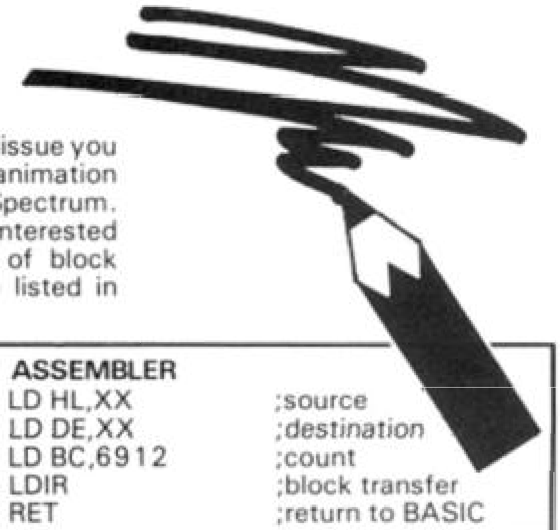
RANDOMIZE USR 0 : resets all the Spectrum, including U.D.G.'s
POKE 23755,100 : the program disappears. . .
POKE 23755,0 : and it comes back

- RANDOMIZE USR 1331 : exploding BORDER, it will stop with the BREAK key.

Stéphane Moureau
Blaton (Belgium)

```
  1 REM TO:ZX COMPUTING
           PROGRAM DEPT.
  5 PRINT                       "
THIS IS A LISTING TECHNIQUE THAT
ALLOWS YOU TO SEE,IN THE LISTING
EXACTLY THE SAME AS THAT PRINTED
ON THE SCREEN.(WHEN THE PROGRAM
IS RUN)"
 100 PRINT                      "
  1
  A SCREEN DISPLAY IS EASIER
  TO PLAN INITIALLY, OR TYPE
  FROM LISTINGS,IF THE PRINT
  COMMAND LINE IS PADDED OUT
  WITH SPACES.(BEFORE QUOTE)
"
 110 PRINT                    ,,"
LINE 120 SHOWS THE SCRAMBLE SEEN
IN MOST PROGRAM LISTINGS. THEY
ARE DIFFICULT,AND EYE-STRAINING
TO TYPE FROM."
 120 PRINT "           2
                  A SCREEN DISPLAY
IS EASIER         TO PLAN INITIALLY
, OR TYPE         FROM LISTINGS,IF
THE PRINT         COMMAND LINE IS P
ADDED OUT         WITH SPACES.(BEFO
RE QUOTE)
           "
 130 PRINT                      "
IT IS BEST TO PRINT THE DESIRED
SCREEN(AS IN LINE 100),AND THEN
RE-EDIT THE LINE AND CHANGE THE
COMMAND AFTERWARDS IF REQUIRED."
 140 LPRINT
"   3
  A SCREEN DISPLAY IS EASIER
  TO PLAN INITIALLY, OR TYPE
  FROM LISTINGS,IF THE PRINT
  COMMAND LINE IS PADDED OUT
  WITH SPACES.(BEFORE QUOTE)
"
 145 PRINT                      "
(LET STATEMENTS CAN ALSO BE USED
AS SHOWN IN LINE 150.)"
 150 LET P$=                    "
    4
  A SCREEN DISPLAY IS EASIER
  TO PLAN INITIALLY, OR TYPE
  FROM LISTINGS,IF THE PRINT
  COMMAND LINE IS PADDED OUT
  WITH SPACES.(BEFORE QUOTE)
"
 160 PRINT P$
 170 PRINT                      "
LISTINGS ARE EASIER TO DECIPHER
USING THIS METHOD,ALTHOUGH EXTRA
MEMORY IS USED."
 180 REM
          ROIS R.HARDER
          995 SHAKESPEARE AVE.,
          NORTH VANCOUVER,B.C.
          V7K 1E7,CANADA
```

Animation



Dear ZX Computing,
In your Feb/Mar 1984 issue you presented a screen animation utility for the 48K ZX Spectrum. Your readers may be interested in a shorter method of block transfer than the one listed in program 1. Here it is:

HEX	ASSEMBLER	
21,XX,XX	LD HL,XX	;source
11,XX,XX	LD DE,XX	;destination
01,00,1B	LD BC,6912	;count
ED,B0	LDIR	;block transfer
C9	RET	;return to BASIC

The first two commands do the same as in the original; setting up the values of the start of the source block and the destination. BC contains the count : the length of data to be transferred and LD1R does all the work. As LD1R takes HL as the source and DE as the destination I have transposed them (as compared to the original version) so that no changes have to be made to the main BASIC program.
Yours faithfully,
Ken Dobson

Countdown

Dear Sir
Being a keen follower of the TV Countdown game I was glad to type in Ben Rimmer's neat program in the Feb/Mar issue.

One thing concerned me though and that was the occurrence of disproportionate numbers of letters like j, k, z, u etc. compared with the commoner letters which can, of course, be regulated with TV games such as Scrabble and Countdown.

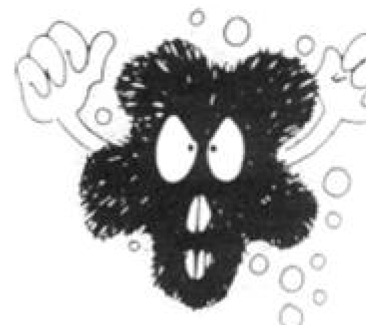
After experimenting I found the following modifications to the program lessened (not eradicated) the disporportions:

Line 20 LET a$ = "BCDDDFGGHHJKLLLMMNNNPQRRRSSSSTTTVWWXYZ"
Line 24 LET b$ = "AAEEEEEEIIOOOU"
Line 40 change (21 * RND) to (38 * RND) + 1
Line 62 change (5 * RND) to (12 * RND) + 1

Yours truly
Derek J Skeet

Plea From The Pennines

Dear ZX Computing
I am begging to you for software for the ZX81. I live in Riddlesden near Keighley and I have been looking everywhere for good games for the ZX81. Pennine TV was suggested so I went along only to find millions of Spectrum tapes but no ZX81 tapes. Could you please, please tell me some good software shops near Riddlesden!?
Yours faithfully
Catherine Robinson



Wall Walk . . . Corrections!

What?. .Not Enough Data?

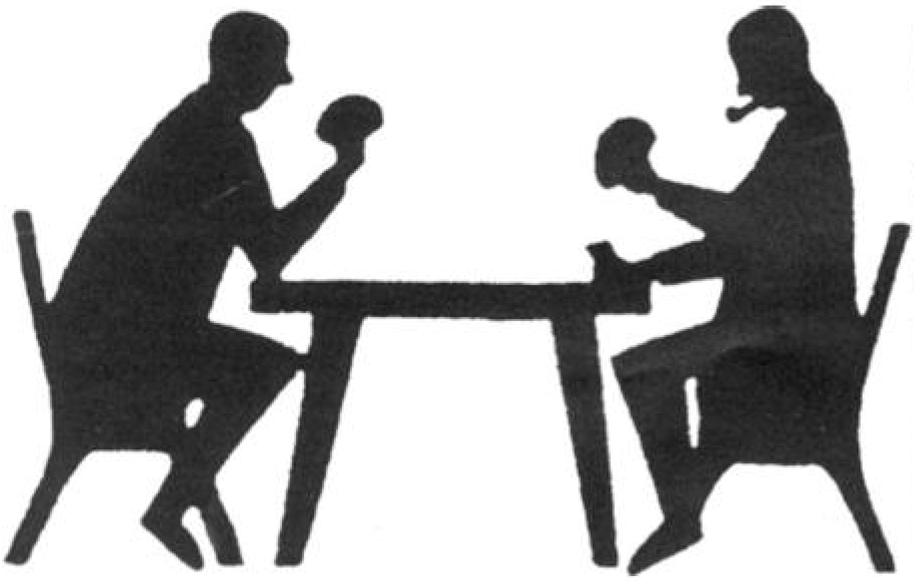
Several readers complained of problems with the Wall Walk program published in the June/-July issue of *ZX Computing*. Readers may wish to cast their minds back to that heady article, which contained two listings; listing 1 to set up the UDG's, and listing 2 which was Wall Walk proper. Listing 1 appears incomplete, as line 10 consisted of a FOR-NEXT loop which was designed to READ 167 data elements from line 20. Unfortunately, line 20 did not contain 167 data-items — guaranteed to cause a hiccup! However, the main Wall Walk program would have run anyway, without the final UDG's being set-up. But, to put readers minds at rest, replace line 10 from listing 1 with the following:

```
10 FOR i=USR "a" to USR
"a"+145: READ x: POKE i,x:
NEXT i
```

Apologies to those readers whose attempts with Wall Walk came to nought.

Anyone for bridge?

Bertil Forss from Switzerland discusses the problems involved in creating a bridge program.



Like some ZX81 owners I am in the process of writing a bridge program. Although the project is far from finished, I assume that my philosophy and methods could serve as ideas for those either in the process of developing a bridge program or attempting any other type of card game.

Aims and restrictions

The primary aim is the use of the ZX81 as bridge partners — where the user is playing one hand and the ZX81 the other three. The restrictions are defined by the ZX81 configuration: ZX81 computer with 16K RAM.

The memory size is a severe restriction as it prohibits the implementation of a desirable playing quality. Considering this, the current goal could be defined as: develop a bridge playing program of moderate quality under the additional conditions of optimising the memory use and achieveing a reasonable waiting time. I am confident on reaching that goal to the price of two deficiences:

The program offers a minimum of dressing as texts need a lot of memory.
The program is compact.

Where considerable saving of memory could be expected, the normal rule of structured programming should give way for more playing power. This has brought a program which is fairly difficult to debug.

The development time is another type of restriction — a psychological one. In hobby programming there are not too many hours available per week and a 16K project could take a pretty long time before the program is available for use. In order to get something useful in a reasonable time I went for a stepwise development where each step was in itself a finished product. The first three steps are (don't worry about naming, in the beginning they were all mnemonics):

KBRIDGE A kernel which could be used as the basic building block. It includes all the basic rules of bidding and playing bridge. The machine is playing like a monkey. This step is finished with 6K memory.

LBRIDGE Expanded to this version the machine is playing at a moderate level. The memory budget was 10K (including KBRIDGE). This step is finished and plays to my satisfaction.

ABRIDGE In this version the machine is expected to bid on a moderate level. It is assumed to gobble up the rest of the memory. This step is in the design phase. Further possible steps are mentioned later.

Data base and presentation

As always at the beginning of a project the inputs, outputs and internal representation should be defined. The internal representation should not be restricted by the display. What is good ergonomy for man is not necessarily the best for the machine. The program therefore contains a subroutine we may call "CONVERT". It builds up a hand on the display from the internal representation. As seen on the figures it can build up a hand in two different ways and it is used for three different pictures:

Bidding phase (Figure 1)
Playing phase (Figure 2)
End of game (Figure 3)

For the internal representation, the string handling capability of the ZX81 is useful. A bridge game contains two important arrays of data: the card deck and the bidding sequence.

The most compact representation of a card deck is 52 bytes, eg called CHR$ 0 to CHR$ 51. Interpret them sequentially as 2, 3, 4, . . . J, Q, K, A of clubs etc and the sorting problem will be easy. After the deck has been shuffled the first 13 cards present the South hand followed by West,

```
     S        W        N        E
             1C        P        P
    3X        D        P        P
     R        P        P        P
END OF BIDDING

            S DEALS

      ALL   IN THE ZONE

      C:Q J 8

      D:10 9 8 3 2

      H:K J 8

      S:6 4
```

Fig. 1

```
    C:7 6

    D:K Q 5
N
    H:A

    S:K Q 8 2
                                  S 3X

                                  NS:1
                     █  C 3
                                  EW:2
    C:Q

    D:10 9 8 3 2

    H:K J

    S:6 4
```

Fig. 2

```
                C:10764
                D:KQ5
                H:A10
                S:KQ82
C:A95                        C:K32
D:7                          D:AJ64
H:9754                       H:Q632
S:J10753                     S:A9
                C:QJ8
                D:109832
                H:KJ8
                S:64
```

Fig.3

S	W	N	E
	1C	P	P
3X	D	P	P
R	P	P	P

Fig.4

North and East. By changing the value, C$, of a card to:

CHR$ (CODE C$ + 65)

A played card can be earmarked but still be retrieved at the end of the game. The bidding string is less complicated as it consists of a concatenation of all the bids in the sequence and the form they are given with two bytes per bid, eg:

''sPsP1HsP2HsP4HsDsRs-PsPsP''

Program structure

Figure 5 shows the layout of the program. SETUP contains the original housekeeping, shuffle and deal and the display before bidding (lower half of Figure 1).

BID (Figure 6) checks that the bidding rules are followed. It either invites South to type the next bid or it jumps to a subroutine for evaluating the next bid. In KBRIDGE the subroutine selects one of three possibilites:

Open with 1 C
Double after 3 of no trumph (3 X)
Otherwise pass

This choice was dictated by program testing requirements. PLAY (Figure 7) sorts out the three cases:

Lead
Follow suit
Void

This checks the basic rules as followed. It invites South (or North if NS has the game) to type the next card to play or it jumps to one of three subroutines, one for each case for evaluating the next card to play. KBRIDGE just takes any card by the basic rules.

TERM displays the result and offers a menu:

A new deal
Repeat the same deal from the bidding
Repeat the same deal and bidding from the play
Display all hands (Figure 3)
Display the bidding (Figure 4)
Rotate the deal one hand and display

The repeat facilities offer the possibilites to try different strategies and proved a good debugging aid. The rotation followed by repeat from bidding allows the player to try his ability on the other hands.

All these functions are included in KBRIDGE. In LBRIDGE the three subroutines for lead, follow suit and void are replaced by more elaborated subroutines. As it is not possible to construct an algorithm for playing bridge, a heuristic approach was taken and appeared as an iteration process:

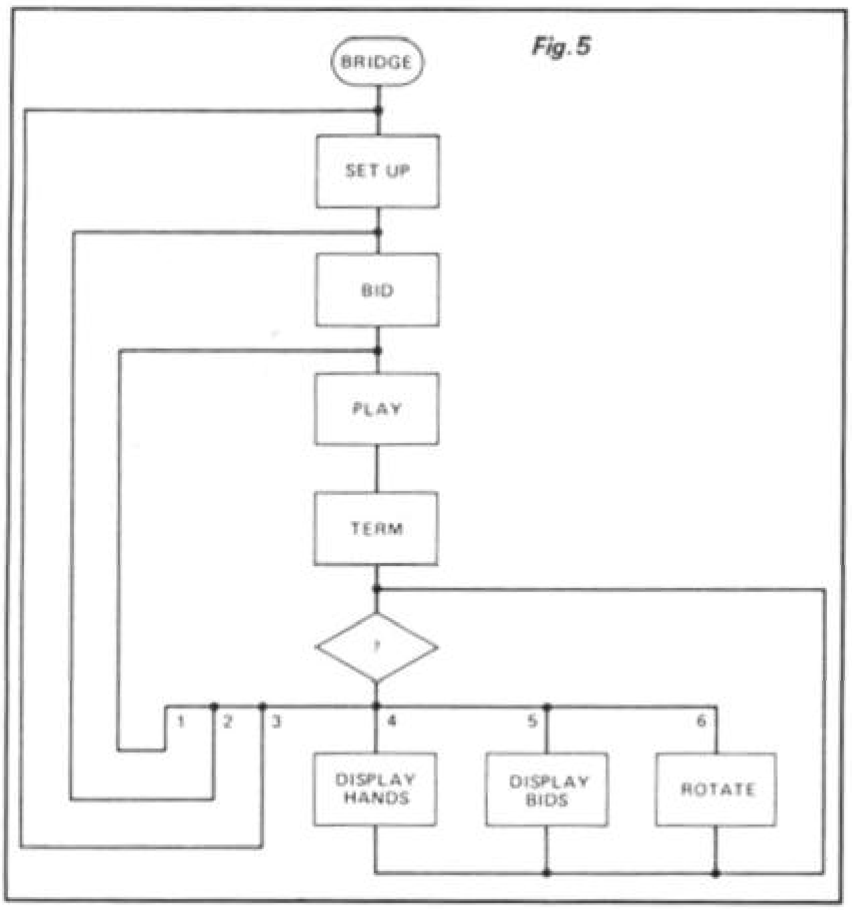
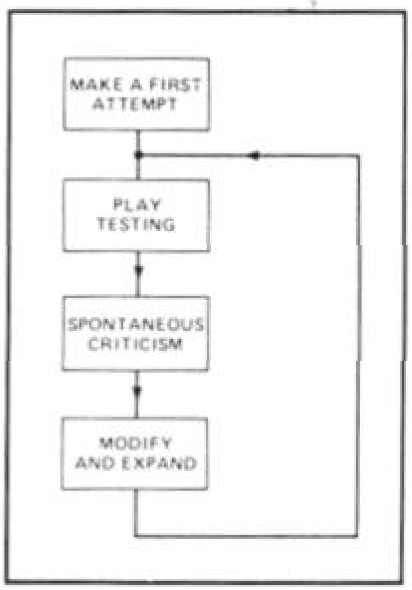


Fig.5



This process was continued until the budgetized memory was filled with program.

In testing this type of program it is important to be alert for any irregularity. A coding errors does not always appear as a failure but more often as a funny playing habit. To trace that habit back to the program error could be very mind bogging.

Some failures can be quite amusing. On one occasion when I thought the program was correct, the machine lost one card and replaced it with a second Ace of trumps! I was quite happy to have the repeat play possibility and could hang on the situation until I had beaten him.

The following might give some idea of the heuristic approach taken. There are hundreds of books written on how to play bridge and it is far from possible to program all those rules into the machine and some of the rules could even be contradictory. So what do you do? My approach was to define seven abstraction levels in two groups:

1 Tactical play
Take the current trick as cheap as possible by using the following information:
a) Played cards of this trick and own hand
b) The hands of the partner
c) All hands

2 Strategic play
Takes as many tricks as possible by using the following information:
a) All played cards and own hand
b) The hand of the partner if open
c) The hand of the partner if closed
d) All hands

Of course, the machine might cheat and look at your cards. So what? The illusion is that the machine is playing fairly well. LBRIDGE includes the three tactical abstraction levels. Strategic play was left for another version as explained at the end of the article.

Some coding examples

It would not help anybody to reproduce the entire listing — it would just fill pages of the magazine and anyone copying it would be lost when tracing a typing error. I have therefore

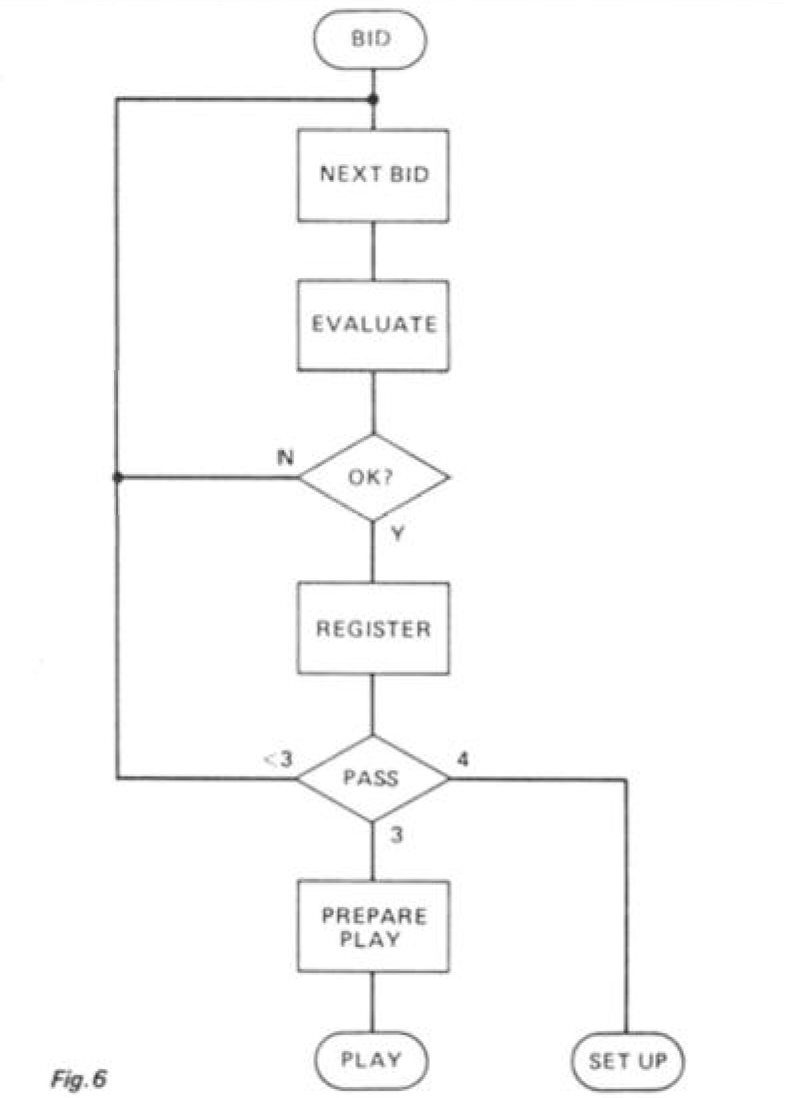


Fig. 6

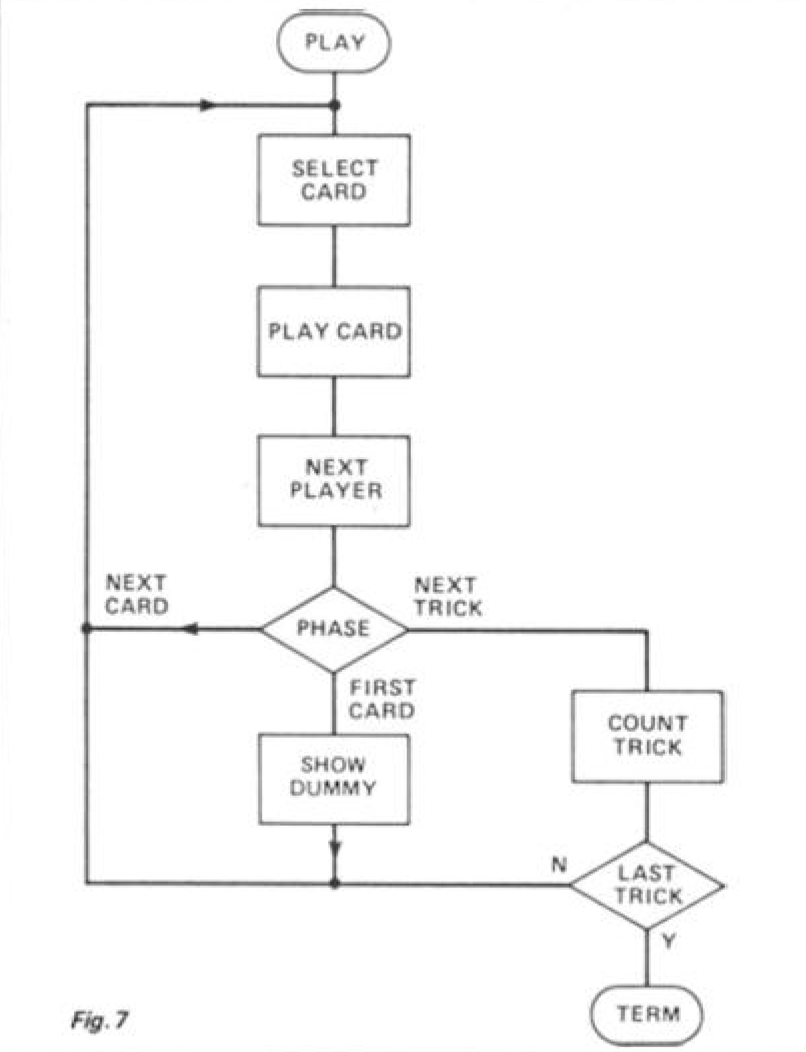


Fig. 7

```
 460 LET K$=""
 470 FOR N=0 TO 51
 460 LET K$=K$+CHR$ (51-N)
 490 NEXT N
 500 FOR N=52 TO 14 STEP -1
 510 LET R=INT (RND*N+1)
 530 LET K$=K$(1 TO R-1)+K$(R+1
TO 52)+K$(R)
 560 NEXT N
 600 FOR M=1 TO 3
 610 LET N=12
 620 LET P=13*M+N
 630 IF K$(P)>=K$(P+1) THEN GOTO
 660
 640 LET K$(P TO P+1)=K$(P+1)+K$
(P)
 670 IF N<>12 THEN LET N=N+2
 680 LET N=N-1
 690 IF N THEN GOTO 620
 695 NEXT M
```

Fig. 8

```
2027 LET E=INT (CODE T$(1)/13)
2030 FOR N=H*13+1 TO H*13+13
2040 IF INT (CODE K$(N)/13)=E TH
EN GOTO 2060
2050 NEXT N
```

Fig. 9

selected two short but typical routines as examples:

Shuffle and deal
Figure 8 shows the code for shuffling the card deck K$. Lines 460-490 fill K$ with cards. Lines 500 -560 shuffle the cards by picking one card at random and placing it at the end of the deck. When 13 cards remain unpicked, the process can stop as these cards all belong to South and they are sorted. Lines 600-695 then sort the cards of the other three hands.

Search
Looking for a card with given qualities is a procedure that frequently occurs. One example, shown in lines 2027-2050 (Figure 9) searches for a card of the played suit T$ (1) in the hand of player H. The suit number (O, 1, 2, 3) is achieved from the card value by dividing by 13. This is compared with the suit number of each of the 13 cards in hand H. The code is bulky but other approaches, for instance, defining K$ as a string array, have disadvantages. If BASIC provided for the redefinition of a string to a string array, this would have been fine.

Future plans

As mentioned, ABRIDGE is now being designed. The current approach is to use a GOTO-matrix with the coordinates defined by the number of given bids and a code defining those having passed. In relation to the abstraction levels in LBRIDGE, a number of ideas exist which could not be programmed in the budgetized memory space. Similar problems will occur in ABRIDGE. Another intersting possibility would be to replace the shuffling by the manual loading a deal, eg from the bridge corner of a journal. The pool of ideas appears never ending — I foresee the following further steps:

BBRIDGE: KBRIDGE + high level bidding
PBRIDGE: KBRIDGE + high level playing
MBRIDGE: ABRIDGE + manually defined deal
CBRIDGE: BBRIDGE + manually defined deal
QBRIDGE: PBRIDGE + manually defined deal and bidding
FBRIDGE: ABRIDGE + scoring for a rubber

In other words, I am looking forward to a number of interesting problems. One of them I would like to mention in the hope of getting some good advice; that is the lack of an overlay feature from cassette to memory. I should like to load BBRIDGE, make the bidding and read in PBRIDGE without spoiling the data. The possibilities I see are:

Saving all data in a common part at the top of the memory
Using QBRIDGE
Buying more memory
Buying a Spectrum!

Toolbox Trio

Mark Wenham shows how to add the ON GOTO construct while Dr Winfried Steinmuller of Austria provides Store and Move utilities — All three for the ZX81.

ON X GOTO / GOSUB

Many computers have a function which will allow you to make a jump or subroutine call depending on the value of a specified variable. However this useful function has not been included in the ZX81's ROM, but the following program remedies this omission.

An untidy and memory consuming set of lines is often to be found in programs which take the form of:

```
1000 IF A=1 THEN GOTO 200
1010 IF A=2 THEN GOTO
     1900
1020 IF A=3 etc.
```

This program lets you replace these lines with two lines as follows:

```
1000 RAND USR 16514
1010 REM ON A GOTO 200,
     1900, 0, 77, etc
```

The variable "A" can be any single letter variable and can have a value between 1 and 255. The GOTO (or GOSUB) is the ZX keyword and is obtained by entering keyword THEN first, using the arrow/shift keys to backspace and then RUBOUT the THEN. The numbers following the GOTO/GOSUB are the line numbers to be branched to — up to 255 numbers can be used, if you can fit them in! Each number must be separated by a comma.

First load a Hex loader program or type in the one given in program 1. The first line MUST be 1 REM followed by 140 of any character, I advise you to LIST 10 before running as some of the code POKEd into this REM may cause the ZX to go into a continual listing loop. Now RUN the loader and enter all the code from program 2. Note that only the numbers in column 1 are entered, column 2 is the address at which it wil be located (in Hex) and column 3 is the assembly code in case you wish to use an assembler or want to follow it through. SAVE your program after all the code has been entered because if you have made a mistake then all your work will 'be lost!

Now enter program 3 which is a test program for your routine, if all works OK (numbers 200 to 600 printed on the screen) then delete all lines except line 1 by entering each line number at a time and pressing NEWLINE. Then save the REM LINE 1 ready for use in your programs.

The program works by using

```
 999 REM (C) SINCLAIR
1000 DIM A$(704)
1010 FOR I=0 TO 21
1020 FOR J=1 TO 32
1030 LET A$(J+32*I)=CHR$ PEEK (P
EEK 16396+256*PEEK 16397+J+33*I)
1040 NEXT J
1050 NEXT I
```

Fig.1 BASIC program from Sinclair ZX printer instructions.

```
   1 REM ***********************
**************************
   2 POKE 16510,0
   3 FOR A=16514 TO 16562
   4 INPUT A$
   5 POKE A,(CODE A$(1)-28)*16+C
ODE A$(2)-28
   6 PRINT A;"    ";A$,
   7 NEXT A
```

Fig.2 Hex loader. Line 1 contains 49 characters. With Move routine, line 1 contains 71 characters, line 3 must be changed to: 3 FOR A= 16514 TO 16584



```
Address  Hex-Code   Address   Label   Mnemonic          Comment
-----------------------------------------------------------------------
Copy D-FILE into A$
4082  CD 9A 14   16514    Start   CALL 5274         ; CLEAR from ROM
4085  01 C3 02   16517            LD BC,707         ; Set LEN A$ +3
4088  F7         16520            RST 30H           ; Expand VARS
4089  2A 14 40   16521            LD HL,(16404)     ; (E-LINE)
408C  09         16524            ADD HL,BC         ;     Set E-LINE/
408D  CD A0 14   16525            CALL 5280         ; STKBOT/STKEND
4090  2A 10 40   16528            LD HL,(16400)     ; (VARS)
4093  36 46      16531            LD (HL),70        ; LET A$ =
4095  23         16533            INC HL            ;
4096  36 C0      16534            LD (HL),192       ; LEN A$ low Byte
4098  23         16536            INC HL            ;
4099  36 02      16537            LD (HL),2         ; LEN A$ high Byte
409B  23         16539            INC HL            ;
409C  EB         16540            EX DE,HL          ;
409D  2A 0C 40   16541            LD HL,(16396)     ; (D-FILE)
40A0  23         16544            INC HL            ;
40A1  06 16      16545            LD B,22           ; No. of Lines
40A3  C5         16547    LOOP-L  PUSH BC           ; Save Counter-L
40A4  06 20      16548            LD B,32           ; No. of Columns
40A6  4E         16550    LOOP-C  LD C,(HL)         ; Ch. from D-FILE
40A7  EB         16551            EX DE,HL          ;
40A8  71         16552            LD (HL),C         ; Put Ch. into A$
40A9  EB         16553            EX DE,HL          ;
40AA  23         16554            INC HL            ; Next Pos. D-FILE
40AB  13         16555            INC DE            ; Next Pos. A$
40AC  10 F8      16556            DJNZ 16550        ; --> LOOP-C
40AE  23         16558            INC HL            ; Skip N/L
40AF  C1         16559            POP BC            ; Get Counter-L
40B0  10 F1      16560            DJNZ 16547        ; --> LOOP-L
40B2  C9         16562            RET               ;
-----------------------------------------------------------------------
```

Fig.3 Hex code and mnemonic listing of screen store routine.

```
Address  Hex-Code   Address   Label   Mnemonic          Comment
-----------------------------------------------------------------------
MC --> RAMTOP

40B3  2A 04 40   16563    Start   LD HL,(16388)     ; RAMTOP old value
40B6  01 31 00   16566            LD BC,49          ; Length MC
40B9  A7         16569            AND A             ; Reset Carry Flag
40BA  ED 42      16570            SBC HL,BC         ; RAMTOP new value
40BC  22 04 40   16572            LD (16388),HL     ; Set RAMTOP
40BF  EB         16575            EX DE,HL          ;
40C0  21 82 40   16576            LD HL,16514       ; Start MC
40C3  ED B0      16579            LDIR              ; Move MC
40C5  CD C3 03   16581            CALL 963          ; NEW from ROM
40C8  C9         16584            RET               ;
-----------------------------------------------------------------------
```

Fig.4 Hex code and mnemonic listing of move routine.

the ZX's floating point calculator to find the value of the variable specified and then looking for the nTH line number in the listing. If the variable is less or equal to 0 or is greater than the number of the following line numbers then the program will stop with error report 1.

Screen store

In the ZX printer instructions, a program is provided to save the content of the screen in A$ (fig. 1). In FAST mode, this BASIC program needs 20 seconds to finish, and in SLOW mode even 'one minute fifteen seconds'! I have written a machine code routine, which does the same work in a fraction of a second. The first part contains three calls to the Sinclair ROM. All variables are cleared with CALL 5274. After loading BC with the required number, RST 30H expands the variables area. This call tests also if there is enough room for those 707 bytes. Call 5280 is used to set the system variables E-LINE, STKBOT, and STKEND. Now the first three bytes in the variables area are set. If you want to use a string other than A$, you have to POKE 16532 with the different code (70 = A$, 71 = B$, 72 = C$ etc.). The bytes from 16540 to 16562 are a direct translation into machine code from the BASIC program in fig. 1. Since this routine contains no absolute jumps, it can be located anywhere in RAM.

I have listed the program residing in REM line 1 (fig. 3). It occupies 49 bytes, and you can load it with the simple hex loader given in fig. 2. Line 2 changes the line number of the REM line into 0, to prevent the routine from being deleted unintentionally.

```
  1 REM (140 of any character)
 10 LET X=16514
 20 LET A$=""
 30 IF A$="" THEN INPUT A$
 40 IF A$="S" THEN STOP
 50 PRINT A$( TO 2);": ";
 60 POKE X,16*CODE A$+CODE A$(2
)-476
 70 LET X=X+1
 80 LET A$=A$(3 TO )
 90 GOTO 30
```

Program 1. Hex loader.

Block move

If you prefer to locate the routine above RAMTOP, the 22 bytes move routine may be of interest to you. Again, this routine is totally relocatable, and can be used to move any code from 16514 on above RAMTOP. You don't have to care about the value of RAMTOP, since it is lowered automatically to the correct address, and after all, NEW is executed. If there is already a MC routine residing above RAMTOP, the move routine won't destroy it, and the code to transfer will be located below the existing code.

For use with the screen store routine, the loader program (fig. 2) has to be expanded as noted in the caption. The move above RAMTOP is then called by USR 16563 (fig. 4). With other routines, the two bytes after LD BC, now equalling 49, have to be POKEd with the numbers giving the length of the machine code routine (eg. POKE n1,232 and POKE n2,3 for 1000 bytes to be moved — and if the routine is loaded behind those 1000 bytes, the move is called by USR 17514).

The screen store program requires at least 3½K RAM, ie. the display file has to be fully expanded. The move routine will work even in a 1K machine.

```
CODE      ADDRESS   ASSEMBLER CODE

2A2940    4082      LD HL,(NXTLIN)
110400              LD DE,0004
19                  ADD HL,DE
221640              LD (CH-ADD),HL
E7        408C      RST 20
FE34                CP 34      "O"
2003                JR NZ, +3
E7                  4091 RST 20
FE33                33 CP 33 "N"
2076                JR NZ,19
E7                  RST 20
CD1915              4097 CALL STACK A
EF                  RST28
2B                  DATA CHR$
1A                  VAL
34                  EXIT
CDA70E              CALL 0EA7
AF                  XOR A
B1                  OR C
2867                JR Z,19
E7                  RST 20
FEEE                CP EE
3062                JR NC,19
FEEC                CP EC
385E                JR C,19
F5                  PUSH AF
23                  INC HL
1E00             10 LD E,00
FE76                CP 76
2856                JR Z,19
23               L1 INC HL
1C                  INC E
7E                  LD A,(HL)
FE1C                CP 1C
3808                JR C,12
FE76                CP 76
280A                JR Z,13
FE26                CP 26
3047                JR NC,19
FE1A             12 CP 1A
3843                JR C,19
20EB                JR NZ,11
23               13 INC HL
0D                  DEC C
20E1                JR NZ,10
2B                  DEC HL
43                  LD B,E
2B                  DEC HL
10FD                DJNZ -3
7E        40D4      LD A,(HL)
D61C                SUB 28d
1D                  DEC E
2823                JR Z,18
87                  ADD A,A
4F                  LD C,A
87                  ADD A,A
87                  ADD A,A
81                  ADD A,C
23                  INC HL
86                  ADD A,(HL)
D61C                SUB 28d
1D                  DEC E
2817                JR Z,18
EB                  EX DE,HL
60                  LD H,B
45                  LD B,L
C5               14 PUSH BC
6F                  LD L,A
29                  ADD HL,HL
44                  LD B,H
4D                  LD C,L
29                  ADD HL,HL
29                  ADD HL,HL
09                  ADD HL,BC
13                  INC DE
1A                  LD A,(DE)
D61C                SUB 28d
85                  ADD A,L
3001                JR NC, +1
24                  INC H
C1                  POP BC
10ED                DJNZ, 14
44                  LD B,H
4F        40FD   18 LD C,A
F1                  POP AF
ED7B0240            LD SP,(ERR-SP)
CA840E              JP Z,0E84
CD1C15              CALL 151C
C3B50E              JP 0EB5
CF               19 RST 08
00                  00 = Error "1"
```

Program 2. Assembler listing of ON GOTO/GOSUB function.

```
100 FOR F=1 TO 10
110 PRINT F;
120 NEXT F
130 PRINT
140 FOR F=1 TO 5
150 RAND USR 16514
160 REM ON F GOSUB 200,300,400
,500,600
170 NEXT F
180 STOP
200 PRINT "200"
210 RETURN
300 PRINT "300"
310 RETURN
400 PRINT "400"
410 RETURN
500 PRINT "500"
510 RETURN
600 PRINT "600"
610 RETURN
```

Program 3. ON GOTO/GOSUB test program.



A maxim which eloquently describes the Brother HR-5.

Less than a foot across, it's nonetheless loaded with features.

The little printer that's low on decibels.

There's one thing the HR-5 won't give you. Earache.

For the annoying 'clickety clack' many printers produce is mercifully absent from the HR-5.

Quietly efficient, it delivers high definition dot matrix text over 80 columns at 30 characters per second (maximum).

Text or graphics with ease.

The HR-5 also has something of an artistic bent. Being capable of producing uni-directional graphics and chart images together with bi-directional text. What's more it will hone down characters into a condensed face, or extend them for added emphasis.

At home with home computers.

Incorporating either a Centronics parallel or RS-232C interface, the HR-5 is compatible with BBC, Spectrum, Oric, Dragon, Atari and most other home computers and popular software.

Perfectly portable, the battery or mains operated HR-5 weighs less than 4 lbs, and has a starting price of only £159.95 (inc. VAT).

Which is really something to shout about.

PLEASE SEND ME MORE DETAILS OF THE REMARKABLE BROTHER HR-5 PRINTER.

NAME __________

ADDRESS __________

____

____

__________ TEL NO. __________

ZXC7/84

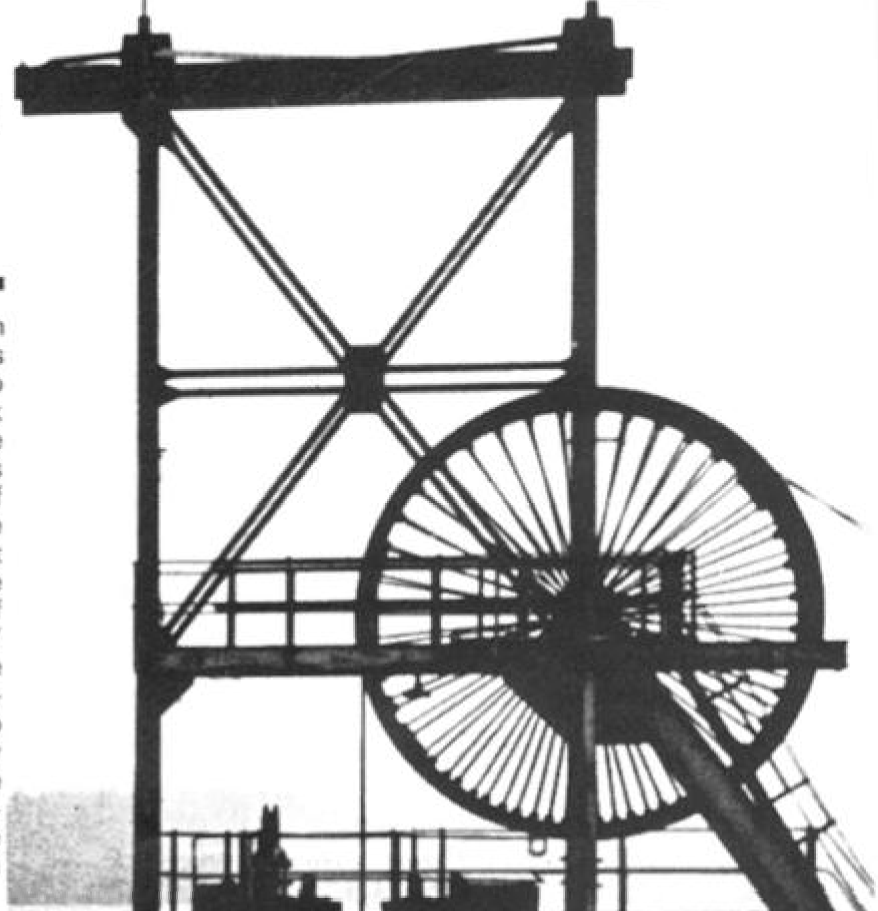
AVAILABLE FROM: BOOTS, RYMANS, WILDINGS, SELFRIDGES AND ALL GOOD COMPUTER EQUIPMENT STOCKISTS.

brother®

DEPT P, BROTHER OFFICE EQUIPMENT DIVISION, JONES + BROTHER, SHEPLEY STREET, GUIDE BRIDGE, AUDENSHAW, MANCHESTER M34 5JD
TEL:061 330 6531 (10 LINES) 061 330 0111 (6 LINES) 061 330 3036 (4 LINES). TELEX: 669092. BROTHER INDUSTRIES LTD., NAGOYA, JAPAN.

Mad Miner

You've got to be mad to work in a plutonium mine say J Dave Rogers and Colin Hogg of Liverpool.



Your job is to collect plutonium (which appears as numbers scattered around the mine) while blasting your way through obstacles and avoiding the acid rain dripping from the stalactites, which are also, of course, inching their way down from the roof of the mine. Any stalactite, when it grows too long, can snap without warning and turn into a falling block. The drips and blocks crash through walkways (black floor) but there are auto-repair systems that will maintain enough for you to walk on (grey floor).

Hazard warning

As long as you have energy you are unaffected by the radiation in the mine, but if you ever let your energy drop to zero then you are destroyed. As if all this wasn't enough, you must also keep a careful eye on the black and white bar-graph across the top of the screen because this represents the running total of *uncollected* points. These are points that have appeared but were not collected and were destroyed by drips or blocks. If this builds up to 'Critical Mass' (all black) then you and the mine are annihilated in a nuclear explosion. This is your incentive to gather up the points reasonably quickly and not miss any! The final hazard is that the doors between the 'lift shaft' and the mine will occasionally slam shut.





How to play

Go under numbers to collect points and under plus signs to collect energy. Each plus gives you another 50 energy units but don't try to go over 200, as 'over charging' will only reduce your energy by twenty units. If you run into or under anything except numbers or plus signs you 'crash', lose fifty energy units and are relocated to the lift shaft.

Control keys for the left hand are 'X' and 'C', which move the man left/right, (key 'Z' can also be used instead of 'X' if you find it more comfortable). Key 'F' fires the laser three spaces in the direction the man is facing at the time, using up twenty energy points in doing so.

Control keys for the right hand are 'M' and 'N', which operate the lifts up and down when the man is on a lift position. These keys are also 'mend' floors when the man is not on a lift position. There are two lift positions (marked by the 'L's) in the mine itself and a five-way-lift-complex in the lift shaft. Lifts stop at walkway levels automatically, not in between, and you can get onto a lift in mid-air (antigravity lifts!).

If you bear in mind that drips and blocks fall through damaged (grey) floors until they hit an intact (black) floor, 'mending' a damaged floor (key M) means that any points underneath it can be protected until you have a chance to collect them. Also, try to leave lifts in the most usable positions. Even so, it is sometimes necessary to clear away 'remains' of previously used lifts by firing the laser into them. If anything is not clear the quickest way to learn is just to play the game a few times.

The program in operation

Line numbering may seem a little eccentric in places but must not be altered as it is due to the use of 'GOTO CODE INKEYS' which routes the program to different line numbers depending on which key is pressed. This gives faster operation than using lots of separate 'IF INKEYS EQUALS' statements for each function, but it also means that pressing keys other than those specified in the playing instructions may give unpredictable results. This is no real problem though as all bottom row keys are OK, and obtaining reasonable speed is the more important consideration with ZX81 BASIC! Despite there being lots of different things happening in the game, man-movement remains at a good pace due to careful overall program structuring.

Along the lines

Here follows a brief description of the program functions, line by line.

5-10	Subroutine called from various points in the main program. Sends program to other subroutines depending on keys that are pressed and what is above the man's head.
50-52	Man move right. Prints man facing right and a space to the left of man to rub out previous position.
53-59	Fire laser three spaces in direction man is facing at the time.
62-69	Lifts up and down stopping only at walkway levels, not inbetween.
73-75	Man move left. Prints man facing left and a space to the right.
200-270	Main loop. 'Grows' stalactites at random and calls all the other routines via line 270.
270	'Goes to' random lines between 270 and 320, thus choosing any one of five randomly occurring events: (drip/print points/mystery bonus points/doors shutting/autorepairs). The line numbers are very important here because they determine the *relative probabilities* of these events happening. Basically, the higher the line number the more chance of it happening. For example, line 275 'shuts the doors' into the mine and has a 5 in 50 chance of being chosen each time the main loop is executed.
320-360	First part of 'drip' from bottom of stalactite.
400-440	Drips/blocks fall through damaged (grey) floors until reaching an unbroken (black) floor. The same FOR NEXT loop is used for dropping either drips or blocks, being set up with different values for each function.
450-480	Decreases level of unbroken floor represented in array H after it has been hit by a drip or block, then reprints floor in grey after it has passed through.
500-590	Print random value points, at random positions, at random times during game. Keeps record of total uncollected points, prints bar graph (line 560) and ends game if bar graph reaches maximum.
600-650	Manual floor repair.
670-672	'Mystery bonus points', 0 to 9.
800-950	Routines for picking up points or energy, then updating and printing the score or energy level on screen.
1000-1060	'Crash' routine, then relocate man to lift shaft.
2000-2080	Auto repairs. Scans along floors repairing every two or three. This is done so that the game can never deteriorate into 'all blocks on the bottom floor'.
3000-	Initialise variables and print screen. Some graphics are held in strings to simplify printing by repeating them.

Typing tips

Line 51	Space, graphic 'greater than'.
Line 74	Graphic 'less than', space.
	It is *essential* to use the above characters, since line 53 looks at these to decide in which direction to fire laser.
Line 250	Graphic shift G.
Line 640	Graphic shift seven.
Line 880	Twenty spaces between the quotes.
Line 2040	Graphic shift seven.
Line 4060	32 greys.
Line 5010	Graphic colons.
Line 5030	24 x graphic shift 7, graphic space.
Line 5040	All graphic: 'L, 8 colons, L, 14 colons, LIFTS, 3 colons'.
	Also check lines 235, 505, 1010.

Before you run the program, erase line 7080 and save a few copies on tape! Note that full on-screen instructions have been omitted to keep program length down but can easily be added. A demonstration — only version just to display the mine/drips/blocks etc can be obtained if line 286 is temporaily erased, line 272 edited to 'GOTO 2000' and line 5 replaced with 'RETURN'. This will then run indefinitely.

Variables

S	Screen position of space above man's head.
X	Marks the start of display file.
L	Level (or line) the man has reached.
M	Man position (left/right).
Y	Character above man IF it is *not* a space.
A	'Across' coordinate for all actions involving stalactite growth, drips or block drops.
FI	Fire direction for laser, depends on character of man.
DI	Lift direction.
H (N)	Height of undamaged floor at position (N) across.
G (N)	Growth of stalactite at position (N) across.
Y	Used to hold value of G (N) to allow faster program access.
ORI, GAP, CH	These set up FOR NEXT loop with origin, gap and character for dropping either drips or blocks.
R$	Random points to be printed.
TM	Total missed, ie points printed but not collected.
BP	Bonus points.
SC	Score.
EN	Energy.
P$	Printing string. Used to simplify initial setting up by repeating. Also used to reprint bottom line after auto repairs have occurred.

Optional extras!

Having typed in and used the main program, the machine code routine can be added. This causes a 'patrol-man' to appear at annoyingly unpredictable intervals and chase you along the levels. Since he's only slowed a

bit by your laser, the only escape is to get to a lift in time!

To add this feature, first type in (as line one) a REM statement containing any 170 characters, then type in and run the hex-loader program (Program 2) and enter the data line-by-line. You will now have to add the lines in Program 3 to your listing.

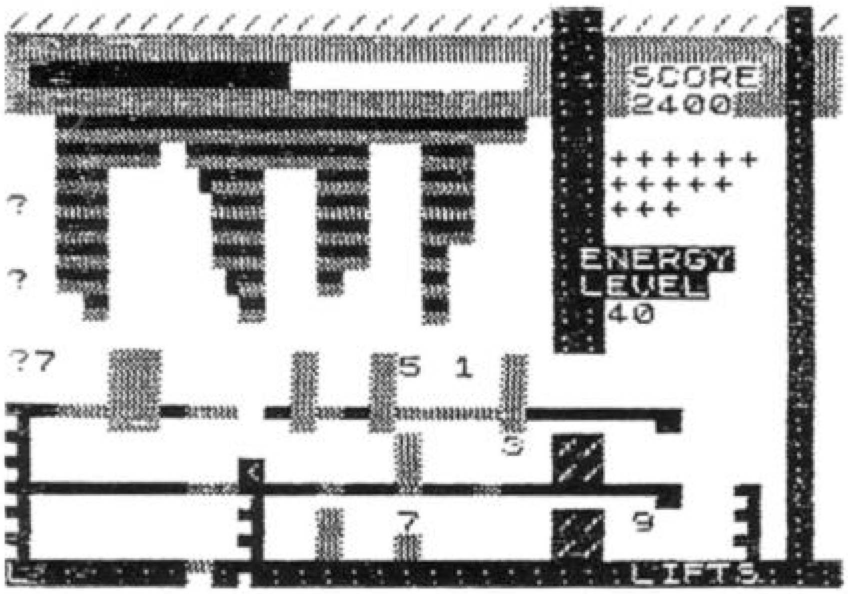
Listing 1

```
   3 GOTO 7030
   4 REM ----SUBROUTINES----->
   5 LET S=X+33*L+M
   6 GOTO 10+CODE INKEY$+(700 AN
D PEEK S<>0)
  10 RETURN
  50 LET M=M+(M<29)
  51 PRINT AT L,M-1;" ";CHR$ 146
  52 RETURN
  53 LET FI=1-(2 AND PEEK (S+33)
=147)
  54 FOR F=(M+FI) TO (M+FI*3) ST
EP FI
  55 PRINT AT L-1,F;CHR$ 134;AT
L-1,F;" "
  56 PRINT AT L,F;CHR$ 6;AT L,F;
" "
  57 NEXT F
  58 LET EN=EN-20
  59 GOTO 910
  62 IF M<26 AND M>0 AND M<>9 TH
EN GOTO 600
  63 LET DI=1-(2 AND (L=20 OR IN
KEY$="M"))
  64 FOR T=1 TO 1+(2 AND L>10)
  65 LET L=L+DI
  66 PRINT AT L,M;CHR$ 146;AT L+
1,M;CHR$ 132
  67 IF DI=1 THEN PRINT AT L-1,M
;" "
  68 NEXT T
  69 RETURN
  73 LET M=M-(M>0)
  74 PRINT AT L,M;CHR$ 147;" "
  75 RETURN
 100 REM ------MAIN LOOP------>
 200 LET G(A)=U
 210 GOSUB 5
 220 LET A=INT (RND*18)+2
 230 LET U=G(A)+1
 235 GOSUB 5
 240 IF U>11 THEN GOTO 491
 250 PRINT AT U,A;CHR$ 137
 260 GOSUB 5
 270 GOTO 270+RND*50
 272 IF SC>2000 THEN GOTO 2000
 275 PRINT AT H(A)-2,21;CHR$ 152
;CHR$ 152;TAB 21;CHR$ 152;CHR$ 1
53
 286 GOTO 500
 288 PRINT AT 7+INT ( RND*3 )*3,
0;"?"
 313 GOTO 200
 320 REM ----DRIPS/BLOCKS----->
 330 FOR N=U+1 TO U+3
 340 PRINT AT N,A;CHR$ 133
 350 GOSUB 5
 360 NEXT N
 370 LET CH=23
 380 LET GAP=33
 390 LET ORI=X+A+132+U*33
 400 FOR N= ORI TO  X+33+A+H(A)*
33 STEP 33
 410 GOSUB 5
 420 POKE N,CH
 430 POKE N-GAP,0
 440 NEXT N
 450 IF CH=23 THEN POKE N-33,10
 460 IF H(A)<21 THEN LET H(A)=H(
A)+3
 470 IF PEEK (X+A+627)<>3 THEN P
RINT AT 18,A;CHR$ 10;AT 15,A;CHR
$ 10
 480 GOTO 200
 491 LET U=4
 492 LET CH=136
 493 LET GAP=99
 494 LET ORI=X+A+297
 495 GOTO 400
 500 REM POINTS/BARGRAPH------->
 505 GOSUB 5
 510 LET R$=CHR$ (INT (RND*9)+29
)
 520 LET TM=TM+VAL R$
 530 GOSUB 5
 540 PRINT AT 13+INT (RND*3)*3,R
ND*28;R$
 550 GOSUB 5
 560 PRINT AT 2,1;" ( 24 inverse
spaces )   "( TO TM/4)
 570 IF TM<84 THEN GOTO 200
 580 PRINT AT 8,1;"YOU MISSED TO
O MANY"
 590 GOTO 950
 600 REM VARIOUS SUBROUTINES--->
 610 IF M>19 THEN RETURN
 620 IF H(M)=15 THEN RETURN
 630 LET H(M)=H(M)-3
 640 PRINT AT H(M),M;CHR$ 3
 650 RETURN
 670 LET BP=INT (RND*9)
 671 POKE S,(28+BP)
 672 LET TM=TM+BP
 800 LET Y=PEEK S
 810 IF Y=15 THEN GOTO 670
 820 FOR T=4 TO 0 STEP -1
 830 POKE S,T
 840 NEXT T
 850 IF Y<28 OR Y>39 THEN GOTO 1
000-(100 AND Y=21)
 860 LET SC=SC+(Y-28)*10
```

```
 870 LET TM=TM-(Y-28)
 880 PRINT AT 2,TM/4; "
           "(1+TM/4 TO );AT 3,24
;SC
 890 RETURN
 900 LET EN=EN+50-(70 AND EN>150
)
 910 PRINT AT 11,23;EN;"  "
 920 IF EN>0 THEN RETURN
 930 PRINT AT 8,1;"ENERGY RAN TO
O LOW"
 950 PRINT TAB 6;"GAME OVER";JD
1000 FOR T=1 TO 11
1010 PRINT AT L,M;"CRASH";AT 14,
24;" ";AT L,M;"     ";AT 14,24;C
HR$ 147
1020 NEXT T
1030 LET M=24
1040 LET L=14
1050 LET EN=EN-50
1060 GOTO 910
2000 PRINT AT 21,5;"AUTOREPAIRS"
2010 FOR N=1 TO 15 STEP 2+RND*2
2020 GOSUB 5
2030 IF H(N)<>21 THEN NEXT N
2040 PRINT AT 18,N;CHR$ 3;AT 15,
N;CHR$ 3
2050 LET H(N)=15
2060 NEXT N
2070 PRINT AT 21,0;P$
2080 GOTO 200
3000 REM ----SETTING-UP------->
3010 LET M=0
3020 LET L=20
3030 LET S=1
3040 LET U=3
3050 LET A=5
3060 LET X=VAL "-32+PEEK 16396+2
56*PEEK 16397"
3070 DIM G(19)
3080 DIM H(19)
3090 FOR N=1 TO 19
4000 LET G(N)=3
4010 LET H(N)=15
4020 NEXT N
4030 LET EN=70
4040 LET TM=5
4050 LET SC=0
4060 LET P$="32 of graphic chara
cter on key h"
4070 PRINT "/////////////////////
/////////////"+P$+P$+P$
4080 POKE 16418,0
5000 FOR Y=0 TO 20
5010 PRINT AT Y,21;CHR$ 142;CHR$
 142;TAB 30;CHR$ 142
5020 NEXT Y
5030 LET W$="24x graphic on key
7"+CHR$ 128
5035 REM * P$ IN NEXT LINE IS  *
         * ALL IN INVERSE.     *
5040 LET P$="l::::::::l:::::::::
:::::lifts:::"
5050 PRINT AT 21,0; P$+P$+P$;AT
15,1;W$;AT 18,1;W$
6000 GOSUB 63
6010 GOSUB 64
6020 PRINT AT L,M;" ";AT 2,24;"S
CORE";AT 5,23;"++++++";TAB 23;"+
+++++";TAB 23;"++++++"
6025 REM * IN NEXT LINE ALL    *
         * TEXT IS IN INVERSE. *
6030 PRINT AT 9,22;"energy";TAB
22;"level";TAB 23;EN;AT 23,6;"ma
d:miner"
6040 LET L=20
6050 LET M=9
6060 GOSUB 64
6070 GOTO 200
7000 REM J.D.ROGERS/C.HOGG,83
7010 CLEAR
7020 SAVE "MAD MINER"
7030 PRINT "MOVEMENT KEYS ARE Z,
C,N,M",,,"F FIRES LASER",,"M ALS
O MENDS FLOORS"
7040 PRINT AT 18,4;"PRESS ANY KE
Y TO PLAY"
7050 PAUSE 4E4
7060 RAND
7070 CLS
7080 RUN 3000
```

Program 2

```
8000 REM
         OPTIONAL
            HEX
               LOADER.
9000 FOR L=1 TO 21
9010 PRINT AT 21,1;"OK---ENTER L
INE ";L
9020 LET X=16514+8*(L-1)
9030 GOTO 9050
9040 PRINT AT 21,1;"ERROR,REDO"
9050 INPUT U$
9060 IF LEN U$<17 THEN GOTO 9040
9070 LET C=0
9080 FOR N=0 TO 15 STEP 2
9085 IF U$(N+1)>"F" THEN GOTO
     9040
9090 POKE X+N/2,16*CODE U$(N+1)+
CODE U$(N+2)-476
9100 LET C=C+PEEK (X+N/2)
9200 NEXT N
9300 IF C+L<>VAL U$(17 TO ) THEN
 GOTO 9040
9400 NEXT L
9500 LIST
```

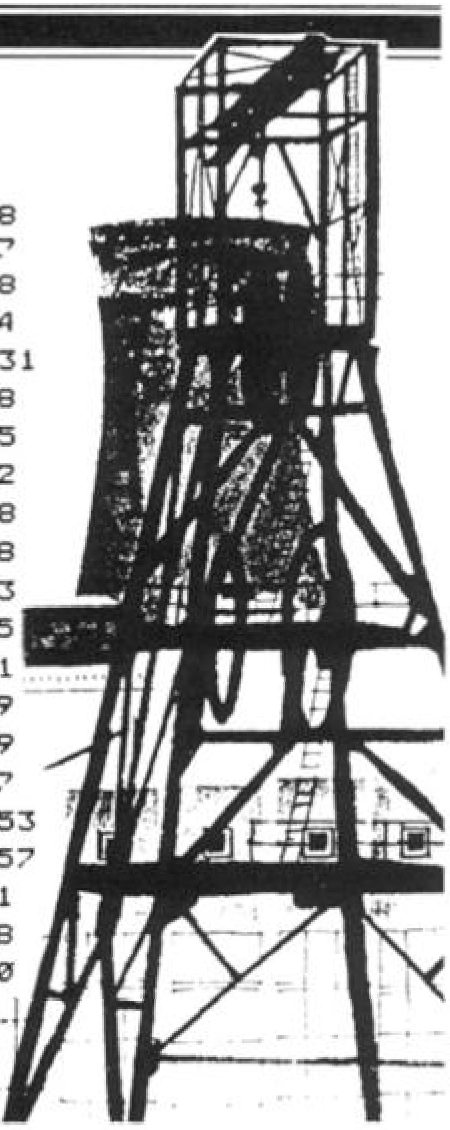



```
   1 REM 17Ø characters of your
choice
  8Ø LET Z=USR 1652Ø
  82 IF Z=1 THEN GOSUB 1ØØØ
  84 GOSUB 5
  86 IF Z<>Ø THEN GOTO 8Ø
  88 RETURN
 273 GOSUB 8Ø
 276 IF SC>17ØØ THEN GOSUB 8Ø
```

Program 3. Extra code which must be added before entering the optional machine code.

Optional machine code Hex dump.

```
ØØØØØ76Ø1ØØ3A823Ø8
4Ø3D28612A1C4ØE5627
D1Ø1ØAØØØ9221C4Ø358
2A1Ø4Ø23Ø1Ø5ØØC5364
EDBØ23C1EDBØCDA71431
ØE51D5CDA7ØED159998
3EØDBA3Ø6Ø3EØ132525
824Ø3EØABBØ1FFFF972
38Ø3Ø1Ø1ØØED43865Ø8
4Ø7A11CEØ1FEØE28728
ØA1131Ø2FE1128Ø34Ø3
1194Ø22AØC4Ø19CB525
78Ø1ØØØØ28Ø3Ø121211
ØØØ922834Ø2A834Ø489
3A854Ø77ED4B864Ø899
Ø97E22834Ø32854Ø627
FE762811F53EB57710 53
F1FE922811FE93281157
ØDØ1FFØØC93EØØ326Ø1
824ØØ1ØØØØC93EØØ478
3285 4ØØ1Ø1ØØC9D168Ø
```



Designs For Living

Computer Aided Design — CAD — has come to the Spectrum. Colin Christmas reveals all . . .

It's not an original prediction but I did reach it all on my own. It's simply this, that in the next few years there will be a rapidly increasing availability of Utilities and 'functional' hardware for home micro users like myself, and of Programs and Add-ons which can be used for educational purposes, for business, marketing, advertising and so on.



Software and hardware which will soon put the small computer like the Spectrum alongside the text book and the audio visual aid on desks at home, in schools, colleges and the office of the small businessman.

I did say it wasn't original and I know I'm not alone in reaching these conclusions. Other natives in the world of the home computer are becoming restless too.

Playing games whether simulation, adventure or arcade poses the same problem as does taking sugar as part of our daily diet. We know we use too much. With that knowledge it's not difficult to cut down or to give it up altogether; "There's more to food and drink than the sugar, lad..." and there's more to a Spectrum or a ZX81 than games.



Does anyone ask a Brain Surgeon to take out tonsils for the rest of his life? Does anyone spend hundreds of pounds on a camera to take holiday snaps? Surely the answer is, "Not if they've got any sense."

What makes sense in this case of the Spectrum and computers like it is that we make the most of its' potential in the most 'user-friendly' ways.

Two important areas being opened-up to home computer users now are to do with graphics and design, both vital to modern technology and Industry. So much so that many Education authorities are now reorganising the traditional and separate basis of subjects like woodwork, technical drawing, art and craft.

The new subject area is being called Craft, Design and Technology. It's an exciting development in educational terms. Teachers of this subject and amateurs like you and I know have access to materials once only available to those who could work with a main-frame Computer.

If you're interested in graphics or design then I can recommend graphics pad, light pens and various 'draw' utilities to open up a new horizon for say the Spectrum. In particular, I can recommend the RD DIGITAL TRACER, from the range of equipment produced by RD Laboratories Ltd.

This is one of many units enabling users to transfer information or images from paper to the TV screen. The least that can be said about such instruments and utilities is that they reveal and demonstrate the potential and capabilities of home computer colour graphics.

There are two versions of the Digital Tracer. The Standard, (£55.50) and the Professional, (£75.50) the main difference as far as I can tell, is that while the Professional Tracer can be calibrated to work from an A3 size grid, the standard tracer can only cope with A4. A software cassette is supplied which contains 5 programs.

The main program, with general purpose trace and draw features, plus 4 others, in BASIC and machine Code loaded automatically to the top of RAM. Programs 1 to 4 are suitable for both 16K and 48K machines and are recorded twice, once on each side of the tape. The 5th program is a composite on side one. You would get the same result by MERGING 2,3, and 4 with the main program. This will only load on the 48K Spectrum. On side two, the 5th program is compiled from Programs 1,2 and 3, and will load on the 16K. The Tracer can even be used with the ZX81 in a memory mapped code.

The ZX81 programs appear as 2 printed listings in the instruction manual which comes with the kit. They will run on the 16K ZX81 using POKE and PEEK to access the Tracer rather than IN and OUT. The first listing offers elementary drawing and printing facilities. The second provides recordings of coordinates for use in analytical programs.

The Tracer itself consists of a short fixed arm and pivot from which extends the drawing arm which is hinged at the centre with a floating pivot joint. This gives a maximum working length of 250cm and an angular range of 0 to 110 Degrees. 0.9 metres of cable connects the arm to the machine via an interface with a 23 way edge connector which plugs easily into the rear port of the Spectrum.

A cardboard template is supplied for positioning the tracing sheet correctly in relation to the arm. The arm has to be calibrated with the grid. After that you're away on your own. The manual is fairly clear and helpful, which is just as well because in order to get beyond the initial thrill of transfering outlines from paper to grid, you're going to need the manual for every next step.

Border, Ink and Paper colours are easily changed. There's a choice of thick or thin lines too. Single points can be plotted and various common shapes — circles, straight lines, rectangles, semi-circles, right angles and so on, can all be simply and rapidly constructed by having to only define two points with the tracing heads. This is achieved by having access to a number of automatic routines.

Complete outlines can be "painted in" or filled with any ink colour. There are two "hatching" effects at your fingertips at the same time. Text, graphics and UDG characters can be added to the screen display as normal, INVERSE or FLASHing. Screens can then be saved on tape or copied on a printer. Previously drawn displays can also be loaded. And thus endeth only the main program.

It's possible then to go on to achieve up and down movement of the display image, movement from side to side, images scaled up or scaled down, reversed left to right turned upside down or any chosen combination of these. As one might expect, the higher the magnification scale used, so resolution and accuracy suffer. Similarly scale reduction of the screen image will produce very fine and accurate detail.

As if all this was not enough, it's possible to produce multiple screen images at different scalings and positions. Finally it is possible to create unique graphics characters which can also then be saved.

The manual boasts that the unit can be used without any programming knowledge. This is true. However, more competent programmers are going to be able to go a lot further. The Tracer's capabilities can be extended by adding BASIC software, additional routines and so on. This way it is possible for graphical information to be used for statistical analysis by using the computer to process co-ordinates of images which have been transferred from originals.

The only criticism of the manual concerns the kit for the Professional model of the Digital Tracer. The additional instructions necessary for the calibration of the tracer arm for the Professional should be incorporated properly in the manual itself and not merely included as a photocopied insert.

The action of the Tracer Arm takes some getting used to as does the fact that rapid flowing movements, when achieved, do not transfer easily to the screen. All movements in fact have to be steady and deliberate. On top of this, it does seem at first to be expensive. With no pun intended, I think that the question of cost needs to be put into perspective and once in use in classrooms, laboratories and office, the Tracer will be the subject of speedy re-evaluation.



One of the areas where it most certainly should have a use is in the introduction to, and then later, experimentation with the techniques of Computer Aided Design. Anyone involved professionally with Electronics, aircraft or vehicle design will have been familiar with this system for many years. But they used big computers to do it, and the whole approach was the province of the experts.

This need no longer be true. And if the RD Digital Tracer seems altogether too daunting and the cost frightens you off, then I can recommend Computer Aided Designer for the 48K Spectrum from Dream Software, as one of several programs now available which will enable you to begin exploring the graphics potential of your machine, seriously and at a price you can afford.

Designed to be user friendly for the beginner, compared to the Digital Tracer, (which would of course be rather unfair), Dream's package (Manual and cassette) can best be described as User Best — Friendly... It's a delight to use. Explanations and descriptions are kept short so that you can move quickly through the manual and familiarise yourself with the program.

The program makes available 27 Commands and the manual takes you through them alphabetically. (thoughtful that!). The only way to do any kind of justice to this utility is to whizz through a few of them quickly.

There are 5 options for designing custom shaped graphics — User Defined Blocks or Sprites. 40 can be constructed, each UDB being 24 by 24 pixels. Circles, parallelograms, rectangles, squares, triangles, can all be drawn to chosen sizes within the given Grid.

3D cubes and cuboids can be drawn automatically (calibrated in units of 16 pixels). Any shapes can be filled with colour, Paper and Ink colour can be changed at any time. The cursor square can be moved all over the grid using the cursor keys and should you find yourself in a corner, then the cursor square can be instantly centred.

Straight lines can be drawn in 16 directions, 26 'User Points' can be assigned anywhere on the screen using letters A-Z and then you can trail lines through all or via selected numbers of them. Lines can be multiplied and joined to a point. Similarly 26 shapes, each containing up to 64 commands can be defined anywhere on the screen. And so on and so on.....

I have to confess to being grabbed by this utility. User friendly indeed. It takes only a short while to learn the 27 commands, and before you know it almost, you are engrossed in the experience as you enter the hitherto remote world of Computer Aided Design.

All screens, sprites and designs can of course be dumped to tape We discovered that screens can also be loaded without first having to load C.A.D. by keying LOAD "" CODE 16384,6912. The potential for designing own title screens using C.A.D. is tremendous.

Already much of this is kids stuff. And that's the point. It's getting difficult to keep up with them. Most start with Graph paper, designing screens, devising their own UDG's. Many have already discovered the disadvantages of trying to work on a 10 by 10 grid and companies like Computer Agencies Limited are now producing PIXEL PADS.

These are large sheets of paper which contain several graphic aids, the largest part being a gridded screen representation, beneath which are several UDG squares.

These squares are an 8 by 8 matrix and also, down the left hand side, the sheets have additional programming information necessary to get started, plus the facility to work in binary in order to enter the graphics as DATA statements.

Not so very long ago, paper pen and ink took over from styluses and papyrus. I don't suppose anyone at the time considered it a dramatic development. It happened because the means was there. Later printing took over from the quill and the scroll. Felt tip pens joined crayons and coloured pencils....

Each successive generation assimilates the facilities enjoyed by the previous generation with startling ease. It won't be long before using the techniques of Graphics and Design will occupy more of our time at home, in school and college when we sit down with our computers.


ANIROG
SPECGRAF
By Nigel Salt
GRAPHIC
DESIGN TOOL
for SPECTRUM (48K)
£9.95
24 x 24 GRID FOR H.R. GRAPHIC DESIGN. 200 CH. STORE. VIEW AND EDIT UP TO 9 CH. — CHOICE OF 9 FORMATS — ROTATE — MIRROR (HOR. OR VERT.) — MENU DRIVEN — KEYBOARD/JOYSTICK CONTROL — TOOLKIT INCLUDED TO USE U.D.G IN YOUR OWN PROGRAMME
TRADE ENQUIRIES: ANIROG SOFTWARE LTD. 29 WEST HILL DARTFORD KENT (0322) 92513/8
MAIL ORDER: 8 HIGH STREET HORLEY SURREY 24 HOUR CREDIT CARD SALES HORLEY (02934) 6083
PAYMENT BY CHEQUE P.O. ACCESS/VISA

Machine Code Plotter

Simon A Rapley has been plotting in Lincolnshire and sends the outcome to us.

Preparing ZX81 Artwork can be a lengthy business in terms of time taken to program a display, time taken for the ZX81 to print such a display to the screen, and memory taken to store the display. This program can make savings in all three of these aspects.

The machine code plotter provided for you is in fact a diagramatic representation of the display file of a ZX81 which has more than 3½K of memory at its disposal. As you can see from this machine code plotter, the ZX81 keeps a record of its display in a string of 792 bytes, every 33rd byte carrying the code for NEWLINE to indicate the end of each row of characters. If a character code is POKEd into any of the 32 bytes preceeding any NEWLINE code, that character will be displayed in that position on the screen. Be warned, however, never POKE anything into any of the NEWLINE codes. Your ZX81 will not like it, and will let you know so! This string of 792 bytes is called the display file, and can be dound in memory by referring to the system variable D.FILE held in addresses 16396 and 16397 (400C and 400D in hex). They hold the address of the byte immediately preceeding the start of the display file.

Try the program LLISTed in Fig. 1. It, like the main program to be explained later, will only run on an expanded ZX81. If you have less than 3½K of memory, the display file is stored differently to make more space available for the program. Can you see why the program in Fig. 1 causes an inverse space to be printed in the top left hand corner of the screen:

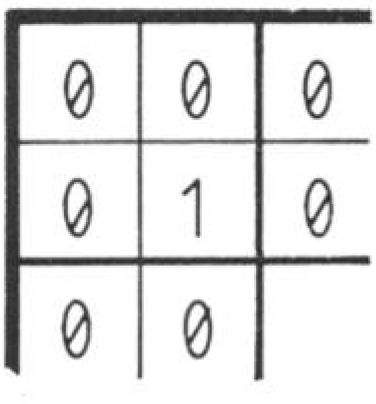
Line 10 reads the system variable D.FILE to find the address of the display file in memory.

Line 20 adds 1 to that address so now it is the address of the first byte of the display file.

Line 30 POKEs the decimal code (128) for an inverse space to that address, so causing it to be PRINTed on the screen.

An extended BASIC program like this could therefore be written to print a picture to the screen. In fact, this would be most impractical, due to the slowness of ZX81 BASIC, and because of the large amount of memory it would require. What is needed is a simple machine code routine that will load a DATA list and print the results on the screen.

Fig. 2 Diagramatic representation of top, left-most character of the ZX81 display



It is easy to see from Fig. 2 that the DATA needs to be assembled in groups of 3 bytes, for each character to be printed. The hex code for an inverse space is 80, and the hex code for the top left position is 0001. The DATA needed to print the inverse space here is 800001. The end of DATA also needs some sort of pointer to tell the printing program that no more DATA is to be read. I chose 43 hex for this pointer as it is the first hex code the ZX81 does not already assign to a character.

All that is needed now is the machine code routine that will read the DATA and do the printing, stopping and returning to BASIC when it reads the end of DATA pointer (43h). Such a program is outlined in the flow diagram in Fig. 3. As you can see, first the character code is read from DATA and checked to see if it is 43h. If it is, the program returns to BASIC. If not, the rest of the DATA for that character is read, its position in the display file is computed, and the character code is POKEd into the correct position. Finally, the program goes back to collect the next character code and start again.

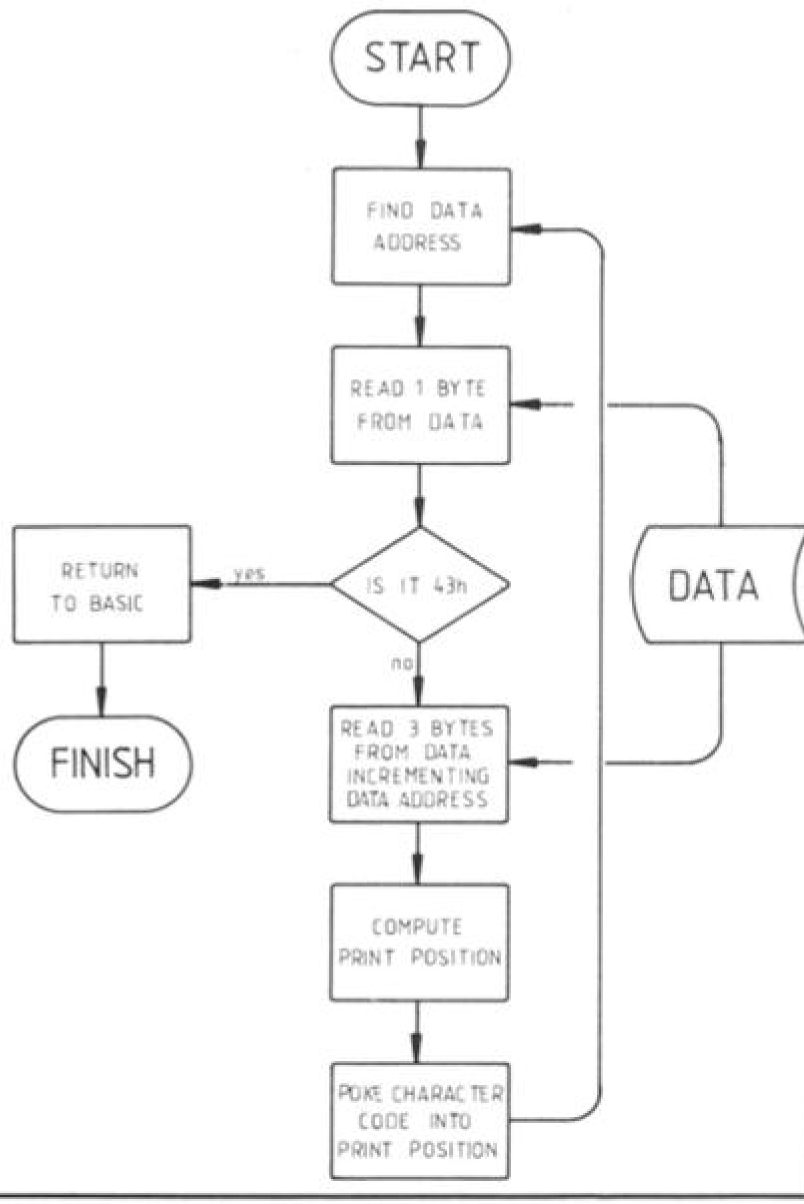
The program, and some DATA for you to try, is given in the print out shown in Fig. 4. The program is also written down the left hand side of the plotter so whenever you use the plotter, you automatically have a copy of the program to go with it. Incidentally, this is an introspective program, in that all the addresses it creates are computed relative to the program or relative to system variables, so that it will not matter which of the traditional places you decide to place it in the ZX81's memory — it will still run properly.

You will soon get used to using the machine code plotter to create pictures of your own design. It is best to mount it on a board, then cover it with clear adhesive plastic sheet, so that you can draw on it with water

Fig. 1 Example of directly accessing the ZX81 Display file

```
  10 LET X=PEEK 16396+256*PEEK 1
6397
  20 LET X=X+1
  30 POKE X,128
```

Fig. 3 Flowchart describing method for loading a character into the ZX81 display-file



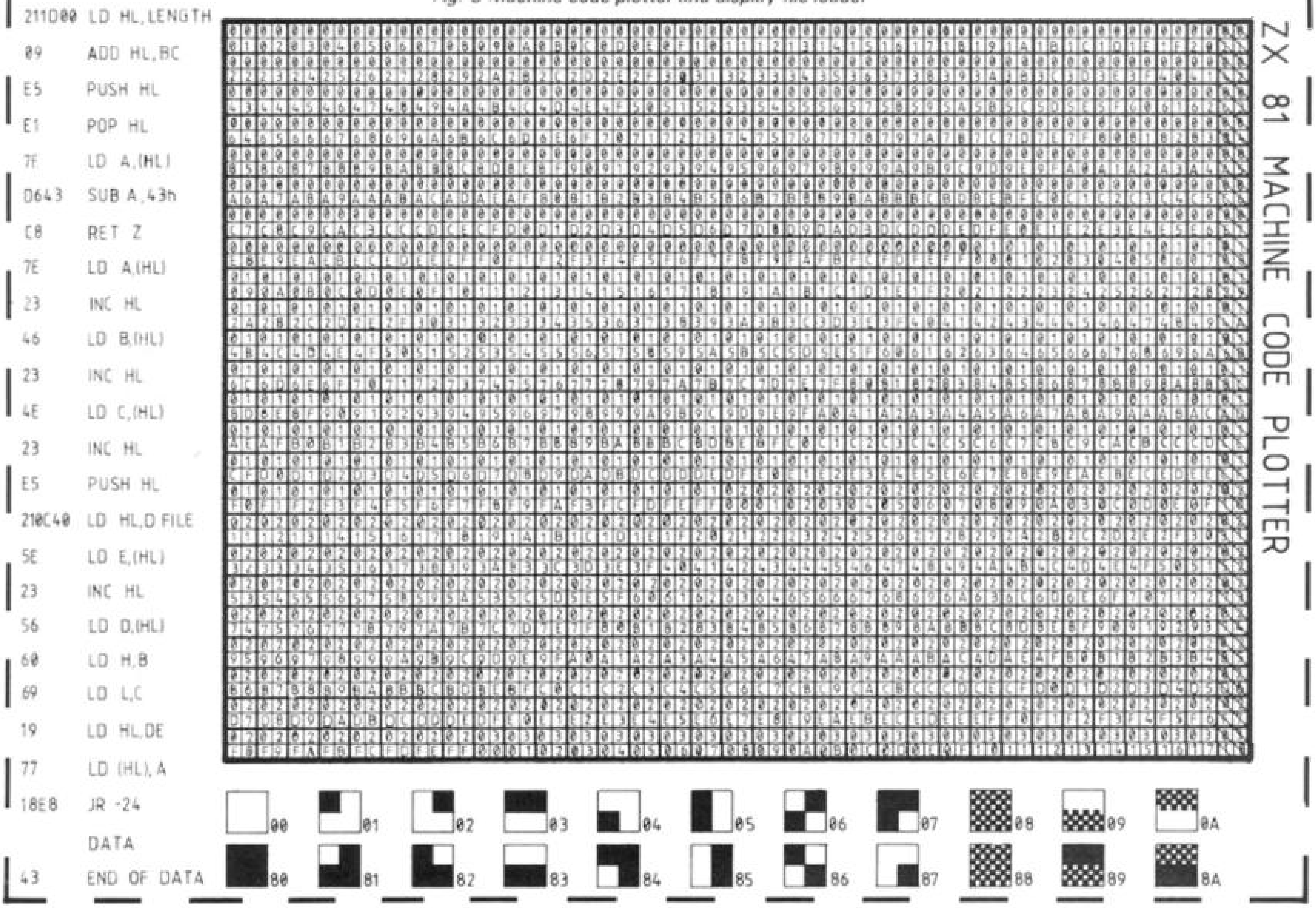
soluble felt-tipped pens, and wipe it clean after use with a damp cloth.
Once your picture is drawn on the plotter, you simply read off the 3 byte blocks as you have drawn them, and enter them into the DATA of your program, remembering to end up with the single byte end of DATA pointer, 43h.

You now have a program and method enabling you to print any design you please to your T.V. screen. So now it's time for you to roll up your sleeves, plug in your ZX81, and get cracking!

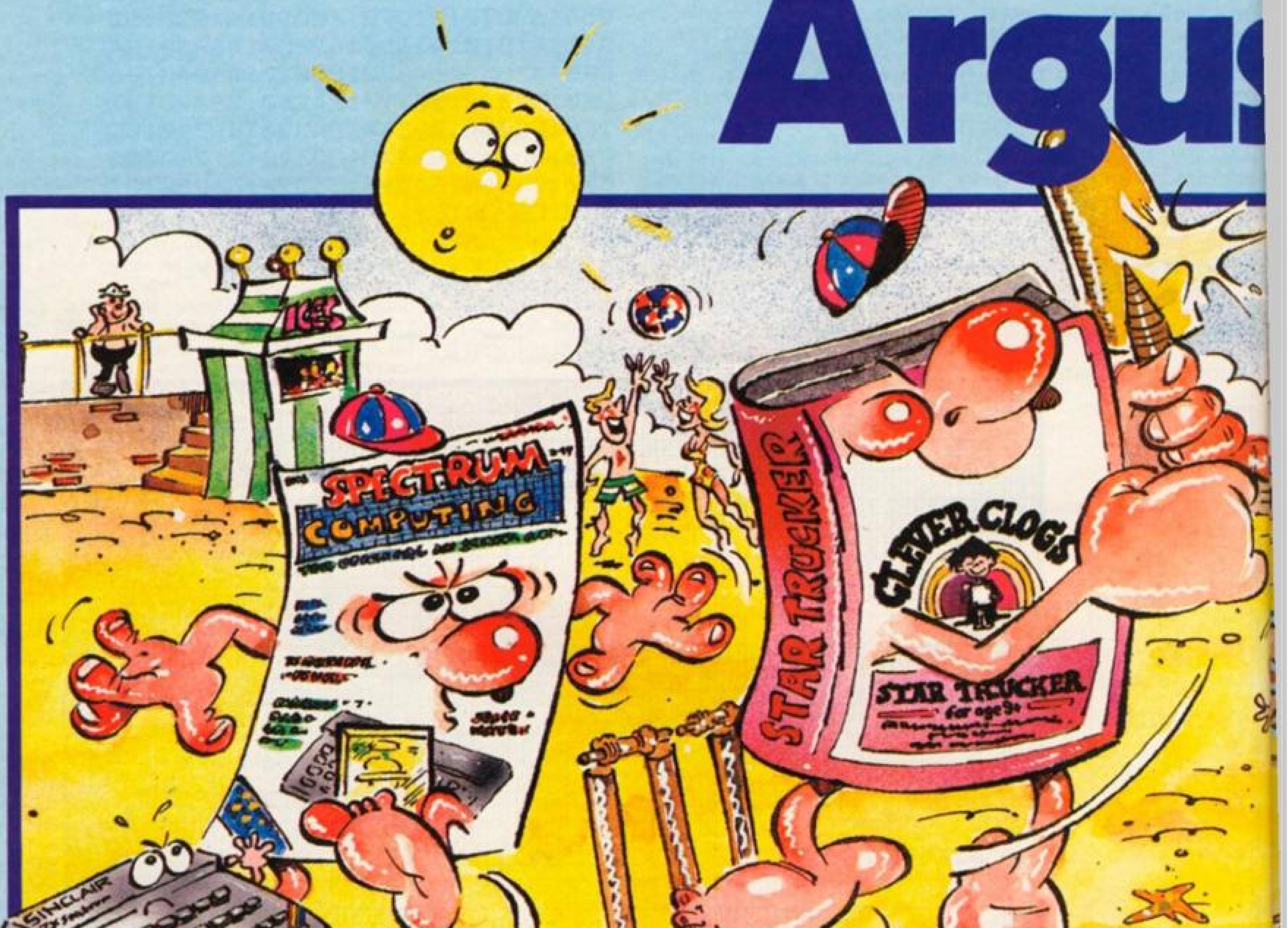
Fig. 4 Machine code and associated assembler listing for display file loader

```
211D00          LD HL,LENGTH
09              ADD HL,BC
E5              PUSH HL
E1              POP HL
7E              LD A,(HL)
D643            SUB A, 67
C8              RET Z
7E              LD A,(HL)
23              INC HL
46              LD B,(HL)
23              INC HL
4E              LD C,(HL)
23              INC HL
E5              PUSH HL
210C40          LD HL,D.FILE
5E              LD E,(HL)
23              INC HL
56              LD D,(HL)
60              LD H, B
69              LD L, C
19              LD (HL),A
18E8            JR -24
07012F840130    DATA
050150830151820171810172850131
030132850152850173830174050133
050154050175070134840135050155
850156820176810177850136850157
850178030137830179040138050159
01017A05013B05015C82017D83017E
85013D85015E85017F830180050l3F
050160050181070140050161820182
840141810183850142850163850184
870143050164020185010144040186
2601B23301B32901B42B01B63701B7
2E01B82A01B93301BA2901BB3101BC
3E01BD3501BF3701C03401C12C01C2
3701C32601C43201C53201C62E01C7
3301C82C01C9     END OF DATA
43               POINTER
5                END OF INPUT
```

Fig. 5 Machine code plotter and display file loader



9 Out of 10
said they
Argus



Tape Magazines:
Great games, great utilities all for only £3.99. Available for Spectrum, Atari, Vic 20, BBC B, and CBM 64 through W.H. Smith, Menzies and computer stores.
Really *use* your computer this summer.

Clever Clogs
8 great fun games for whizzkids. Change the difficulty with data packs on Science, The Arts & General Knowledge.

On sale in W.H. Smith. Keep your Clever Clogs busy all summer long! Available for Spectrum at £6.95.

Mind Games:
Recently launched, the new name in quality strategy and adventure games. Top writers and programmers open the door to the world of imagination and fantasy.
Have an adventure this summer with Star Force Seven, Quest for Eternity, and Orion Quest. Spectrum £7.99, BBC B, an CBM 64 £9.99.

Fall of Rome
Five million barbarians, an army or two of Persians, several thousand upset Armenians, fifty legions of itinerant Italians . . . . . . and you!
Have a holiday at home and save civilisation. Available for Atari, BBC B, Spectrum & Commodore 64.

Computers* preferred Software

Argus Press Software Group



CB, and

two of

it

CB,

The House for Heroes!

The Game Lords join the Argus Press team. Consistently in the top ten with great games like Ant Attack, Boogaboo, Fred, Mined Out and Snowman. Prices from £4.95 to £7.95. Available from W.H. Smith, Boots and Menzies and leading Computer Stores worldwide.

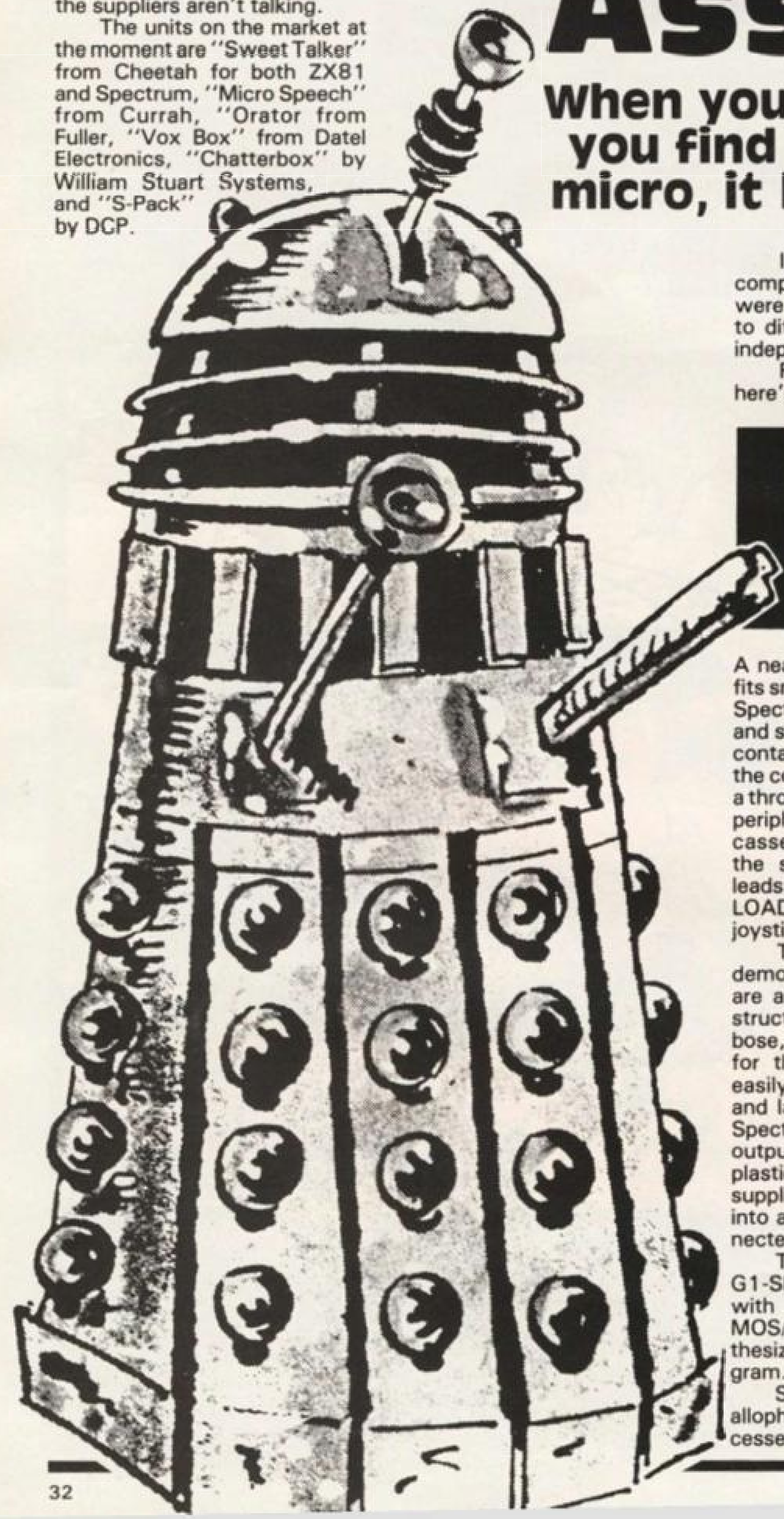
*When asked to express a preference

Talkback: Speech Units Assessed!

When you get to the stage where you find yourself talking to your micro, it helps if it answers back!

From the time the Daleks trundled across our TV screens muttering "Exterminate, Exterminate" in their chillingly inhuman voice, I have been intrigued by speech synthesizers. At the time of writing, and to the best of my knowledge, there are six different units on the market to enable your micro to answer you back. Two others have recently lost their voice, I asked why, but the suppliers aren't talking.

The units on the market at the moment are "Sweet Talker" from Cheetah for both ZX81 and Spectrum, "Micro Speech" from Currah, "Orator from Fuller, "Vox Box" from Datel Electronics, "Chatterbox" by William Stuart Systems, and "S-Pack" by DCP.



I contacted all of the above companies and those units that were supplied for review I sent to different reviewers for their independant comments.

For your own assessment here's what they thought . . . .

ORATOR
Fuller Micro Systems
71 Dale St.
Liverpool
Price £34.95

A neat black plastic box which fits snuggly onto the back of the Spectrum and is identical in size and shape regardless whether it contains the Orator on its own or the complete Master Unit. It has a through bus allowing all the ZX peripherals to be attached, a cassette interface enhancing the signal and allowing both leads to be left in place when LOADing and SAVEing, and a joystick interface.

The unit is supplied with a demonstration tape and what are alleged to be complete instructions. These are too verbose, complicated and ill defined for the novice to understand easily. With its amplifier and larger speaker it gives the Spectrum a controllable audio output, although this requires a plastic screwdriver as no Knob is supplied — a metal one turns it into a radio! It can also be connected to your Hi Fi system.

The speech bit of it uses the G1-SP025A speech processor with a single chip N-channel MOS/LSI circuit that will synthesize speech via a stored program.

Speech is created by an allophone system which is accessed by utilising port 159. In practical terms each allophone is operated by sending numbers with OUT 159,n — where n is the appropriate numeric value assigned to that sound.

The unit is complex to program, step by step simple instructions would have helped, and it has the annoying habit of being incompatible with some commercial programs causing a crash or system reset or irritating unstoppable hiss.

The quality of sound can be very good, clear and distinct, but I'm afraid the programming takes up too much time for me to make frequent use of it.

Tony Woods

SWEET TALKER
Cheetahsoft Ltd
24, Ray Street
London EC1R 3DJ
Price £29.95

The SWEET TALKER unit (11 × 7.5 × 3cm) plugs easily and firmly into the user port at the rear of the computer and has its own rear card edge connector for a Ram Pack necessary. A demonstration program cassette is supplied and not only gives a good idea of the range of speech sounds possible but also explains how to program the unit and even gives a few ideas for turning games and original programs into 'talkies'. It takes less than a minute to load.

One way to understand how spoken words are formed is to break up the sounds that combine to make a word into allophones, and this unit uses the allophone system to produce words. For instance, the word Cheetah can be broken down to 4 speech sounds or allophones: /CH//IY//TT//AA/.

Using the allophone system, each allophone, pause or gap in speech (as at the end of words and sentences) is turned into a number and then given a code. The code is then incorporated within a very simple program using OUT Instruction to 7.

An instruction booklet is supplied with the unit and lists the allophones with everyday words which contain examples of the relevant allophones. Find the allophones required by checking through the word lists, then read across to the column listing the Data Codes and enter them in the first statement, followed by four more short lines and RUN. It talks . . . . .

There are several bonuses with this system. Firstly, although experimentation is necessary to achieve maximum clarity, it's great fun and often amusing. The booklet is easy to use. Secondly, it teaches the user a lot about the structure of words, vowels, consonants, word endings and so on. Not unlike the use of phonics in the teaching of reading. In this last respect this system could well find a place in classrooms. One word of warning to parents and teachers; the unit will say *anything* it's told to — and without so much as a blush!! But when will we be able to program our micros to sing "Daisy, Daisy . . . "?

Colin Christmas

S-PACK
DCP Micro developments
2 Station Close
Longwood
Norwich
NR13 4AX
Price £29.95

Unlike most speech units on the market today the DCP Speech unit does not create words nor does it use allophones.

The unit contains a pre-recorded chip which contains about 70 words which cannot be added to or changed.

The words have been created by first having been spoken and then changed into a digital form, they can be recalled by the user by simply using an OUT statement. ie: OUT 255, 34. The advantage of this is that you get a clear reproduction of sound unlike the allophone system.

You can buy another 3 chips which contain another 70 words each, but will cost you another £12.95 each. These are just slotted into place inside the box. The DCP speech unit will fit the ZX81 and the Spectrum, and fits straight onto the output port.

You can also fit an external speaker if you wish, which can be adjusted by a volume control. The chip that comes with the unit contains the alphabet plus numbers from 1 to 20, and also the words hundred, thousand, million. So in fact you can construct any number you wish.

The other three chips contain words like control, equal, flow, gallon, high, inches, kilo, limit, lower, stop, emergency, pico, kilo and so on. This gives a total of about 280 words. Using the unit is very simple and it does add a new dimension to your programs, although you can bet the one word you want won't be available.

The unit itself costs £29.95 and if you were to buy the unit and all the chips it will cost you nearly £70, need I say more?

Clive Smith

CURRAH MICRO-SPEECH UNIT
Currah Computer Components Ltd.
Graythorp Ind. Est.
Hartlepool,
Cleveland.
Price £29.95

This unit is one of the most easily obtainable, as many shops including Woolworths sell it. Packed in an eye-catching box, the unit itself is neat and lies flat on it's back when connected to the user port, unlike most peripherals which stand upright. It is sold complete with a well written manual and demo tape which also contains an adventure game featuring speech.

A useful feature is that the sound, both speech and the normal Spectrum sound, are output through the TV speaker. A drawback is that the unit is dead ended and you cannot add anything on after it. I also found that the leads which are permanently attached to it are a little short if you want to add it on after other units. The unit users an allophone system, but instead of the usual technical means of using the OUT instruction followed by the respective number, Micro Speech reserves S$ for its own use and a ROM in the unit decodes the required word.

Although the system is more related to English words, to get the correct pronunciation you have to break the word into its component syllables and reconstruct it in Currah language. "Hello", for instance, is spoken by using LET S$ = "HE(LL) (OO)" And of course there are ways to add information and inflexion to the speech, all of which make it a reasonably complex bit of equipment to use.

Currah have spent a great deal of time and effort persuading many software houses to adopt their unit for producing speech in their games, and this is obviously an incentive when purchasing such a unit. Most games also tend to use the Kempston joystick interface as a standard, and the two units cannot be operated at the same time, at least until now!

Currah once more show that they are aware of the market and it's shortcomings and have produced an extension to the Spectrum's port which will allow two (or more) units to be used simultaneously. This can be used with other items such as printer interfaces of course.

After all this, how does it perform?

To be honest, I was dissappointed at first. It didn't seem to live up to its promotion. The demo tape seemed indistinct and words were not pronounced clearly at all, a bit of very careful adjustment of the TV and the unit improved this slightly.

However after using it for a few days I became used to the electronic voice and began producing scathing comments to go with my games. The software companies who use it seem to get a better quality of speech and so far those who did use it used it sensibly and sparingly.

I must now admit that I find it an interesting and effective extension to my Spectrum and will make much use of it. Recommended to anyone who has an interest in going one step further.

Ray Elder

CHATTERBOX II
William Stuart Systems
Quarley Down House
Cholderton
Nr. Salisbury
Wiltshire SP4 0DZ
Price £39.00 + VAT

Chatterbox II is quite a large unit — 10 × 15 × 6 cms — and is housed in a black plastic case which has two DIN sockets and a mini jack socket fitted on the top.

The two DIN sockets allow you to connect the WSS Speech recognition unit (£49.00 + VAT) to one, and either the output from the WSS Music Synthesiser and 16 line control port (£19.50 Kit, 25,00 built, + VAT), or output to an amplifier to the other. The mini jack is an input from the Spectrum's ear socket to amplify its beeps.

Compatible with BOTH ZX81 and Spectrum, it is attached by a cable and plug similar to that on the ZX printer, this means that though the port is OK for ZX81 peripherals, very few Spectrum units will fit on the back.

Speech is achieved by using phonemes or individual speech sounds, and WSS themselves admit that the quality is not as good as pre-programmed speech, but add, and I see no reason to disagree, that the flexibility is infinite.

A list of these phonemes is given in the 5 page booklet which is brief but reasonably clear providing you are an experienced user. To allow for connection to other machines a lot of technical detail is given, but unfortunately not how the INPUT/OUTPUT socket is wired so I couldn't try it with my Hi Fi. To actually produce speech you have to break the word into its component sounds, look up the appropriate numbers and output them to port 127 using OUT 127, no.

The quality of sound depends on the amount of time that you spend in getting the phonemes right, even at its best it still sounds electronic, but clear and recognisable.

This will give hours of pleasure if you like tinkering, also it has great potential educational value as word structure has to be fully understood to get good results!

Two complaints, first the loudspeaker is open at the front of the unit and could easily be damaged — surely it would not have been too hard to have covered it with a mesh or even cloth, and secondly there is no volume control. My family have banned me from using it when they're watching TV.

Finally if you are a fan of Whizz Kids and admire the computer system used by them, then WSS go a long way towards providing a system with practical possibilities — fascinating!

Jim Watson

news

In. . . . . .Brief

• Mirrorsoft added two new educational games to their range for the 48K Spectrum and priced at £7.95.

Well packaged in library cases LOOK SHARP contains two programs from the highly respected Widget Software firm and aimed at the 3-7 and 6+ age ranges.

COUNT WITH OLIVER is a series of early learning maths exercises which use humour and the ability of the Spectrum to encourage 4-7 year olds.

• Cloud 9 Software, 4 Dolvin Rd., Tavistock, Devon, PL19 9EA specialise in Chemistry programs and have added REACTIONS to their Titrations and Formula packages. Two simulations, a collision model and revision notes are included for £8.50.

• Homestudy produce two sets of programs for 'O' Level revision. Physics and Mathematics (two separate syllabus sets) are extensively covered in twin cassette packages for £22.00 and £30.00 respectively.

• Expensive, but so sure of the value of their programs are they, that your money will be refunded if you fail your exam. Find out more from Treleigh Woods, Treleigh, Redruth, Cornwall.

• S.D. Micro-systems, PO Box 24, Hitchin, Herts. SG4 0AE release CALORIE CONTROLLER (£5.95) and PERSONAL FILING SYSTEM (£6.95) for the 48K Spectrum. If you are interested in Domestic/Home Business type programs then contact them for a full list of their wares.

• Fowler Software, Hendon Mill, Nelson, Lancashire have reduced the price of their 48K Spectrum CARTOON ANIMATION program from £10.00 to £5.00.

• GARDEN BIRDS, a program from E H Harbard, The Old Customs House, Sheriffs Brae, Baniff. for the 48K Spectrum is not recommended to Ornithologists! — Who says? Mr. Harbard himself.

It is intended for use by youngsters and those beginning to become interested in birds — I'm sure there's a joke there somewhere.

Teaching without tears

Always a consistant supply of educational software, Chalksoft Ltd. of 37 Willowslea Rd., Worcester, WR3 7QP have recently added to their range with a variety of 48K Spectrum programs. NOTE INVADERS, as the title implies, is an aid to recognising music notation using an arcade game format. Also included on the tape are two other programs which follow the more usual question and answer sequence.

PIRATE is an adventure game. Adventure games are useful as part of the so called "hidden curriculum" in that they encourage logical (and lateral) thinking, creative stimulus, and group cooperation. Also compass and map work etc. is often required. DAS SCHLOSS and EIFFEL TOWER are two language programs. These are question and answer exercises with each section based around a topic or "family" of related words. The main advantage these have is that the user can key in his own word lists and so increase the difficult of the vocabulary as his ability improves. A picture is built up as a visual rewards. All four of these cost £9.25.

Latest in the range is a multi choice (four options) quiz game which is packaged in a large presentation case with a booklet. Personally I think that it is over promoted for the program it encloses, that is not to say that the program is poor, in fact it is a very competent game.

HOTLINE could have been sold in the usual format, the booklet would have fitted onto an extended cassette insert, and perhaps the price of £11.25 could have been reduced. Best described as a quiz database and questionmaster it comes with 10 sets of questions and the facility to create and save your own sets. A visual reward of the "hotline" telephone link being repaired on correct answers gives this program its name. The questions supplied are varied in content and difficulty and have been well written. A useful program for school or home and I agree with the age range quoted — 8 to 80!

Old buffer — new venture



Three years ago Buffer's first Sinclair software shop opened and since then they have advised many programs authors on how and who to send their work to.

In an adventurous (in more ways than one) move, they have decided to market under their own name any good programs that come their way, and the first four arrived in our office. All our supplied in library cases and are, so far, of the non arcade type.

SUPER BRIDGE is intended to give above average players a good run for their money yet contain concise instructions and rules of play to enable the beginner to use it.

MUSIC is brilliant. It can only be used by owners of a sound box which uses the AY-3-8912 chip, ZON, FULLER, and TIMEDATA are three examples, an the OUT number must be known.

It virtually provides a three channel recording studio with editing facilities. Probably the best music composing aid on the market for the 48K Spectrum, almost worth buying a sound unit for. They tell me that it was inspired by the Quicksilva BBC program and this is one I've used and admired, now Spectrum owners can enjoy the versatility of a similar program.

The remaining two programs are adventure games in the traditional text only format and both were written with the aid of the ubiquitous QUILL adventure language program.

BUFFER ADVENTURE features the Buffer shop personal and locations — at least at first — and a dangerous but humorous lot they seem to be too!

Finally FAIRGROUND is the most complex of the two and much more mysterious.

With the greater use of professional programmers by many of the established software houses any outlet for the gifted amateur is to be welcomed. I'm sure we all wish them the best with this venture.

In Brief — Adventures

• THE CODE from Soft Concern Ltd. is described as "A new era in adventure games'. Written for the 48K Spectrum and styled like a spy thriller there is a £2,500 prize for the first person to crack it.
It is priced at £9.95 and is available from 49 Albert St. Warwick.

• "The last word in thinking men's computer games" is how Beyond Software describe what they call the world's first 'epic' game called THE LORDS OF MIDNIGHT. A multi character graphic game, it has a claimed 32,000 views and is obtainable from Durrant House, 8 Herbal Hill, London EC1R 5EJ. Tel 01-837 3699.

• THE CITY OF EHDOLLA is the first in series of standard text only adventures from Goldstar Software. Well packaged, it will provide a challenge to the adventurer, I played it for a brief 30 mins and couldn't get past the first problem.

• M C Lothlorian, established producers of adventure and strategy games have released "Special Operations" in a new warmaster series. For the 48K Spectrum (what else?) and priced at £5.95, I tried it out and was most impressed. A combination of strategy and adventure, the combinations of actions to be tried must be incredible. I wish that I had more time to spend playing it! It is obtainable from 56a Park Lane, Poynton, Cheshire SK12 1AE.

• Serin — new Bridgemaster — have produce the follow up to the Bridgemaster Learning Package, favourably reviewed in an earlier issue, called Introduction to Expert Bridge.
The reason for the change of name to Serin Software is that they are developing programs in a wider range of subjects, meanwhile this new one will cost you £12.95 from PO Box 163, Slough.

• Incentive Software Ltd. 54 London St., Reading RGf1 4SQ are now selling the second in the series of the Ket Trilogy called TEMPLE of VRAN, if you've completed the first and are ready to tackle this one it will cost you another £5.50.
MILLIONAIRE is another release from this company and is a simulation game, run a software house and drive a Rolls.

• WAXWORKS is the latest from Digital Fantasia of 24 Norbreck Rd., Norbreck, Blackpool, Lancs. FY5 1RP.

• Postern has PSY-SPY which starts with an arcade sequence and then runs into an adventure game. For the 48K Spectrum at £7.95 from PO Box 2, Andoversford, Cheltenham, Glos. GL54 5SW, it also includes sequences of speech if you possess a Currah speech unit.

• Phipps Associates, respected produces of Knight's Quest and Pharoes Tomb now bring us COLDITZ ADVENTURE, another standard format text and graphic game.
With some interesting and useful developments (eg. TAKE ALL) it costs £6.95 form 172 Kingston Rd., Ewell, Surrey KT19 0SD.

• SWORD AND SYTHE is a historical simulation/strategy game from Nemesis Productions, Arlee (Wales) Ltd. Neath Abbey Rd., Neath, Glamorgan, and costs £5.75.





Nosweh strikes again

These Seiddab just won't give up!

Well and truly thrashed by legions of purchasers of Hewson Consultants Ltd. two previous games 3D Space Wars and 3D Seiddab Attack, they return for more in 3D LUNATTACK.

In 16K there is an impresive series of sequences and graphics which are even better than its predecessors. Currah Micro-Speech is also utilised to great effect.

The game is a fairly complex 3D perspective variation of the Defender theme, fight your way through waves of attackers to attack the Command Base.

Well worth finding a store to demonstrate this program if you like space zap games. The price is £7.95.

At the other extreme is FANTASIA DIAMOND, a 48K Adventure game. Your fabulous Fantasia Diamond has been stolen, Boris the masterspy who was sent to recover it has been captured, so who is the fool — sorry, hero — who will have to recover it and rescue Boris? You guessed it, YOU.

Some scenes are depicted graphically but the game is essentially text, there are a great many characters, some friendly, some decidedly not. You will have to sort out the goodies as without help you've no chance!

As usual there is something special about a Hewson game, and this one has a powerful language analyser which will allow the player to use fairly complex sentences with several commands to a verb.

One for the fanatics at £5.95.

Morwood computer datacorder

The Morwood Datacorder is one of the latest computer cassette recorders to appear on the market. It incorporates a futuristic styling and advanced function controls. It is attractive, compact and lightweight. The Datacorder has all the normal functions e.g. save, load, rewind, fast forward, stop and eject. Also, according to the manual, it has a pause key, which was so neatly concealed that I was unable to find it.

Morwood have included an auto-stop mechanism for safety, just in case the user forgets to stop the tape once the program has loaded. A slide volume control is situated to the right of the function keys. The slide control makes the recorder more slimlined but it is sometimes less accurate than standard volume controls. They have also included a three digit tape counter so that you can take note of where your programs are located on a given cassette.

It has a transparent cassette compartment which adds to its attractive look. The user can also hear the program being loaded but this can be muted by moving the monitor switch to the off position. on the side of the cassette recorder next to the monitor switch are two essential jack sockets, a rem socket and a din plug socket. The two essential sockets are the input and output sockets. Inside the Datacorder is an automatic cut out circuit which gives the user the advantage of not having to change over or disconnect lead when saving or loading a program.

The Datacorder comes complete with a separate 6 volt power supply unit which plugs into any normal household socket but, if you prefer, batteries can be used. The Datacorder takes four 1.5 volt batteries but, I don't personally recommend the use of batteries as when they run low a program cannot be saved or loaded successfully. The power left in the batteries is indicated on the LED indicator. If the LED is bright, the batteries are satisfactory but, if it is dim, they should be replaced.

An additional attachment is



available for an extra £3.00 which will match the load/save signals to the ZX81 by boosting the signal during saving, and the Editor tells me that he tried it out and it worked perfectly for the brief time he had it. Although the head azimuth still needed to be adjusted occasionally on some tapes the signal tone and strength were perfect.

Overall I am very impressed with this piece of equipment, loading and saving has been 100% successful, and I recommend it to all Spectrum and ZX81 users.

Available from:
Meow Micros
8 Newnham Close
Braintree
Essex CM7 7PR
Cost £24.95 (£27.95 ZX81)

Scott Lintern.



In Brief — Arcade

• CP Software, 2 Glebe Rd., Uxbridge, Middlesex UB8 2RD have taken on that excellent pinball game from Saggitarian and marketed it under their name.

SUPERCHESS 3.0 is claimed to be the "strongest yet for the Spectrum". It has a wide variety of options and a user manual, but no rules of the game.

SUPERCODE is their tool kit offering, and it contains 100 machine code routines. This is the most comprehensive unit I've ever seen and includes all the usual Renumber type programming aids plus 35 screen routines.

Suitable for both 16K and 48K Spectrums, the only problem I can see is that it is definitely an experienced users program.

AIRBASE INVADER is the first satirical arcade game I've seen. A sort of "wander around dodging others and collecting things" type game, its nonetheless quite addictive.

The Characters are cute and humorous, you control Peace Woman Annie — complete with CND flag — around Greenham Common airbase avoiding various other thinly disguised public figures. A nice touch is that your "weapon" is blowing misses and the only one who is impervious is the "Iron Lady".

• Avalon Software, 62 Albion St. Mansfield, Notts is a new company to me and they have three programs on the market.

OMADAN ATTACK is a space battle type game, BOSSMAN is a business simulation and PIT STOP is an arcade driving game.

I haven't tried any of them but I'll get our reviewer to look at them for a future issue.

• Carnell Software still haven't produced the Wrath of Magra at the time of writing, but I've seen the accompanying "Book of Shadows" which is an impressive tome. I can't wait to see if the program lives up to it! Cost will be £12.50.

• Salamander Software finally got Jeff Minter's MATRIX and METAGALACTIC LLAMAS BATTLE AT THE EDGE OF TIME into the Spectrum and, while not state of the art graphics and animation, good solid stuff. Each cost £6.95.

Meanwhile the Dan Diamond trilogy is available as a set with bumper illustrated case file for £14.95. 17 Norfolk Rd., Brighton, E. Sussex.

• Virgin recently produced SORCERY, arcade adventure and ATLAS ASSIGNMENT, adventure game both at £5.95 — they get better all the time.

• Kuma is selling FROG SHOOTERS, an arcade game from the same Japanese company — Hudsonsoft — as the latest Sinclair releases. It is available at £6.95.

• DK'tronics, a consistantly good company, has let loose TROM, a simple but of the "Just one more go — I KNOW I can do better" genre arcade game and ZIG ZAG, a 3D perspective maze game with Currah speech.

• Cheetahsoft leap into the Spectrum software market with two games at £6.95 each. Conquest, a grid type tactical game and 3D Bat Attack, a maze game. I'll get our overworked reviewer on them as soon as possible.

• Visions is getting fruity with THE REVENGE OF THE KILLER TOMATOES for the 48K.

• A superb graphic game form Gargoyle Games in the space battle field. AD ASTRA is one you really must try out at your local shop. Its the large size cartoon type graphics which make it special.

£5.95 from 4 NW Arcade, Birmingham, B2 5LH.

• DISCO DAN will lead you a dance if Gem Software has anything to do with it. Unusual and worth a look, its a cross between run and jump and Q'Bert.

• Only two ZX81 tapes this time, surely the suppliers haven't quit already!!

Bits and pieces

Well, the QL has started to make its appearance and opinions differ widely as to just how good it is. I have heard buyers claim it has a "few minor" bugs and others say it has major faults!

Only time and the eventual modified version will tell, meanwhile I wonder if the Spectrum is about to be neglected in the same way as the ZX81. How long has it been, I ask, since you saw an advert for any Sinclair ZX goods from them?

Having written this and committed myself to paper, during the time it takes to get from page to print they will probably launch a major publicity campaign! Still, at least they have marketed a new set of software titles.

Printer Poleaxed

Many times in my association with the ZX printer I have felt like reducing it to its component prices, but somehow it seemed sad when Sinclair Research announced that they were discontinuing the manufacture of the beast.

Noisy, unreliable and often unreadable, it was nevertheless yet another breakthrough enabling thousands of users to get a cheap introduction to the benefit of hardcopy.

The many letters I've had over the project to find the best list producing system showed that getting good printouts is a priority for many ZX81 and Spectrum owners.

Many people will be grateful that the Alphacom 32 is still available.

USA Abandoned

Now that Timex has deserted the market in the USA, I have received a lot of letters asking what help those owners of the TS2068 will be getting with programs and peripherals.

My answer has been pretty unhelpful. Phone calls to Sinclair produced a "sorry we'd like to help but..." reply I haven't got and can't get hold of a TS2068 (I would even pay to get it modified for the UK TV system!) and no-one I've contacted seems to have any knowlege of the ROM/Peripheral compatibility. If any hardware — or even Software — companies out there **know** that their products are compatible with the USA machines then please let me know and I'll publish a list in a future edition.

Micronet Fishing

The Micronet 800 system which provides a communication, information and telesoftware service on Prestel has made several improvements to their service.

The full range of microcomputing services offered by Prestel is now available at no extra charge, initially the only apparent change will be that the opening page will read "Prestel Microcomputing incorporating Miconet 800".

Prizes

The Sinclair Prize for fiction was not, as some cruel journalist claimed, won by them for the delivery date promises, but b Edward Fenton for his nove "Scorched Earth". Sir Cliv himself said it is a novel "of higl literary merit combined witl comtemporary social anc political relevance".

Sinclair Research won a priz in the form of the accolade fron company directors in the Britis electronics industry as the com pany which has "most enhanc ed the image of the industry as a whole".

They will send a congratu latory note within 28 days.

• 1K ZX81 GAMES is supplied by Iain Thompson of 25 Wallridge Drive, Holywell, Whitley Bay, Tyne and Wear, NE25 0NL. It comes with an instruction sheet and cost £3.95.

• And for the 16K user FLIGHT is a flight simulation from J. Paton, 2 The Avenue, Fallings Park, Wolverhampton, WV10 0RG. The price is £4.95 and I'll get Nick Pearce to have a look at them.

Come on ZX81 commercial suppliers, there's room for you to get your programs reviewed here....



Sinclair Software for the Spectrum

Five new ZX Spectrum arcade games, specially written to test users' reflexes and speed of action, involve exciting contests of skill and chance with a variety of determined and 'fiendish' adversaries.

The programs are 'Eric and the Floaters', 'Stop the Express', 'Bubble Buster', 'Driller Tanks', and 'Zipper Flipper'. The first four were produced by Tokyo based, Hudson Soft, the last by R.E.D./Sunshine.

In 'Eric and the Floaters', the relics of a lost civilisation, hidden deep beneath piles of dust and rubble wait to be discovered and Eric aims to find them by blasting away the rubble with his bombs, but he must also overcome the deadly 'Floaters' who live among the debris.

To 'Stop the Express', the player must run the length of a twenty-carriage train to find the key which will allow him to stop the train. On route he is chased by gangsters determined to prevent him from reaching his destination.

The only way to avoid being squashed by bouncing bubbles in 'Bubble Buster' is to fire at them with a ray gun. However, once hit, the bubbles split in half, and will divide again if shot at a second time. Only when they have been split for a third time, does the player succeed in bursting them.

In 'Driller Tanks' fire breathing Mammuts and their accomplices, the Skorks, try to invade the summer palace and only the tanks can stop them. The tanks must drill through a tunnel complex beneath the palace and then overpower the deadly creatures by freezing and crushing them.

'Zipper Flipper' is a fast moving pinball and fruit machine game in which winning on the fruit machine will greatly boost the pinball score.

All the games have full colour, high-resolution graphics with sound. Priced at £5.95 each. inc. VAT, and supplied in cassette-form, they are available from Sinclair Research Limited, Stanhope Road, Camberley, Surrey. Tel: 0276 685311, and selected retail stores.

Tax-Calculator Software

Sinclair Research has introduced a personal tax program, the 'Which? Tax Calculator', for its ZX Spectrum computer.

Developed by the consumer association it will reckon accurately and automatically the true amount of personal tax owed for 1983-84 tax year. Users are taken step-by-step through the annual tax form and will find the task greatly simplified as a result.

Specially written to include

all relevant basic UK tax rates and allowances, the program is divided into three sections.

The first defines personal circumstances and job earnings, including such items as overtime payments and fringe benefits. Section two contains questions about additional income such as pensions, social security benefits and investment income.

The last covers personal allowances and outgoings — for example mortgages, maintenance and covenant payments. Once all this information has been typed in, the user's true tax rate is calculated automatically.

Prices at £12.95 inc. VAT and supplied in cassette form complete with a 59 page reference manual, 'Which? Tax' is available by mail order from Sinclair Research Limited, Stanhope Road, Camberley Surrey. Tel: 0276 685311 and through selected retail outlets.

Company info



What's in a name?

Doric Software has found the similarity of its name to that of the Oric computer to be too confusing so they have change their name to DORCAS Software.

The name is derived from DORrell Computer Adventure Software and is indicative of the type of programs they produce. Following in the footsteps of Oracle's Cave, a unique D&D graphics style game, is "The Runes of Denos" which will also be along similar lines and have twelve separate but linked adventures. The game has the following features:

Monster animation
Problem Solving
Sound effects
Sentence input
Performance evaluation
Microdrive facilities

Educating Argus

The publishers of *ZX Computing*, Argus, have developed the software side of the company by creating Argus Press Software and have linked with the family firm Computertutor to market their expanding Cleve Cloggs range of educational programs.

Even though they were only a small family firm they had been getting good reviews from the magazines and were respected in the educational field.

Party Time — 3+, Jungle Jumble — 5+ and Whizz Quiz — 7+ were their first titles and have recently been followed by Music — 7+ and Star Trucker — 9+, the last actually making an appearance in the software charts.

The standard also improved as they gained experience and this reflects in the success of Star Trucker. Blockbuster is a simple idea which proves to be frustratingly difficult and is the latest release as we go to print.

I had the pleasure to meet Peter Holme recently as he was slaving over a hot computer putting the finishing touches to another program and I was impressed with his flair for inventiveness and professional presentation — Blockbuster could be another blockbuster!

In . . . . Brief

• At the last ZX Microfair I went to I had the pleasure of seeing two Disk drive interfaces working and demonstrated.

P2L or Primordial Peripherals Ltd. showed me a complicated but impressive system aimed at the small business owner. Priced at £249.95 with a drive unit (£79.95 for the interface) it was fast and efficient, although it needed a bit of study to get to grips with the operating system. It used 2K of RAM for the DOS and a further 6K workspace.

Technology Research Ltd.'s system however was simple to grasp as it used many existing keywords and only an amazing 128 BYTES of RAM. Priced at around £85.00 I have asked them to send me one for a full review, but as yet one hasn't appeared.

P2L is at 89 Herne Rd. Bushey, Herts. WD2 3LP.

Tech. Res. Ltd. is at Unit 18, Central Trading Est. Staines, Middlesex, TW18 4XE.

• Tape Recorders can be a problem and many new ones with "computer compatibilty" are making an appearance.

A purely functional, plain (if not downright ugly) machine is on sale from Coomber Electronic Equipment Ltd. Croft Walk, Worcester, WR1 3NZ.

However the pure functionality of it is stressed as it is intended primarily as a piece of educational equipment. As one who has suffered form "absenteeism" due to it being borrowed and once stolen, I can value the off putting plainess of the casing. At £69.95 it is unlikely to be bought for the home, but when you consider that many people will spend hundreds on fitting a printer/monitor etc. It does seem strange that the main storage media is often neglected.

More normal in appearance is the NW 608 from Network Marketing Ltd. Sutherland House, London NW9 7BT. With sliding volume and tone controls and automatic stop and level control there appears from the press, release nothing outstanding about this machine.

Cost £27.95.

• Keyboards are becoming more varied in appearance, three very nice units have recently made their debut:

Lo Profile from Advanced Memory Systems Ltd. Woodside Technology Centre, Green Lane, Appleton, Warrington WA4 5NG at £49.95 uses a new idea in keys by having them operating on a rubber bubble instead of the normal spring system. A numeric keypad and space bar is built in, and there are a total of 53 keys.

I got a quick look at the Fox Electronics new keyboard which is small and neat and contains a built in amplifier with speaker, volume and tone controls.

Woking Computer Centre, 32, Chertsey Rd., Woking, Surrey have a slimline keyboard with 28 special function keys and at first glance it looks quite impressive. It is priced at £59.95.

DK'Tronics also have a new version of their keyboard on the market, hopefully remedying the design problems of the old one. I found the earlier version good, and I have always found them to be a considerate company to deal with.

Ricoll make a very sturdy metal cased keyboard which also looks tidy, it is pretty basic but does contain a space bar and will probably last forever. Most important it is widely available at around £39.95.

With this wealth of keyboard around I'll get some full reviews in future issues if, and when, the manufacturers supply a review model.

• DK'Tronics have produced two interesting looking units. There is a programmable joystick interface which has a full through port, something essential if you want to use speech units or other interfaces at the same time, and a light pen and interface.

Priced at £22.95 and £19.95 respectively these units look to be good developments and additions to your Spectrum.

• For those interested in CAD (Computer Aided Design) British Micro. 0923 48222 have a unit called Grafpd which looks like a drawing block and pen. This is likely to find a market among schools and enthusiasts only as it's priced at £143.75 inc. VAT — more than the Spectrum itself!

With all this new hardware the final problem must be where to keep it!

But regardless of the connection of the company to *ZX Computing,* I will continue to give unbiased views and send programs to Mike Edmunds for his independent opinion.



You're never too young

Malan Associates was formed in November 1983 to create and produce a complete range of educational, arcade and adventure software, primarily for the Spectrum.

In early April 1984 the Company retained the talents of Andrew Green, a 15 year old programmer who is still at school studying English, maths, computer studies, biology and business studies.

Andrew's first programs launched by Malan were "Gold Blast", a 48K Spectrum arcade game £5.75 and "Character Generator" £8.00, a very complex graphic program that allows computer users to create beautiful graphics on a scaled enlargement basis and on the reverse side a "Drawing Program" allows users to draw any amount of displays and pictures and save them for their own program creation.

In late April Andrew wrote "Music Maker" £6.00 and "Computer Challenge" £4.95 which were launched at the 12th ZX Microfair at Alexander Pavillion with great success "Music Maker" is designed as a utility to give computer users the facility too, once having created and saved a computer program, they can add full sound effects or musical tunes to the program, enabling the adventure or game to become alive with full sound effects. "Computer Challenge" is a family computer game written to test your wits against the computer. Based on a well known game, it gives you options of words, shapes, numbers and word association to challenge the computer, and gives you seven options to beat the computer.

Within a period of six weeks Andrew has produced four programs which have been well accepted by both trade and retail, within the U.K. and Europe.

All these programs are now available through retailers or direct from Malan Associates. Just call Malan's sales office and they will tell you where you can obtain these new and exciting programs.

With Andrew's continuing programs, plus Malan's educational, adventure and arcade range of programs, they offer over 11 programs right now and expect to be increasing their range by 4 to 6 programs per month.

With retail prices ranging from under £5.00, their slogan of "Affordable Software" does seem to be true.

DGH Software Centre, 10 North St., Ashford, Kent

Situated in the heart of Ashford, the DGH Software Centre will be stocking educational, business and other serious software, as well as an impressive selection of games. The centre will also stock a choice of the popular home computers, printers, disk drives, cassette machines, monitors and all sorts of add-ons together with a wide range of books.

The Centre will also run educational projects where a small number of students can have an evenings intensive hands-on training on a specific aspect of computing. Planned courses include 'Advanced Basic', 'Introducing Machine Code' and 'Parents and the Micro'. These sessions will be with either Commodore 64 or Spectrum computers which will be provided for the evening.

Director, Christopher Denne, heading the project, states: "We aim not just to sell computers and software, but also to help people get more satisfaction from their computers by understanding them better."

"Because the staff of the centre are all computer enthusiasts they are always willing to help a fellow user."

In . . . . Brief

• At last, now you can plug in two (or more with extra units) of those interfaces that are designed as dead enders!

Currah, the company who makes the popular Micro Speech unit, have produced a little device which plugs into the port at the back of the Spectrum and allows you to attach two interfaces or units.

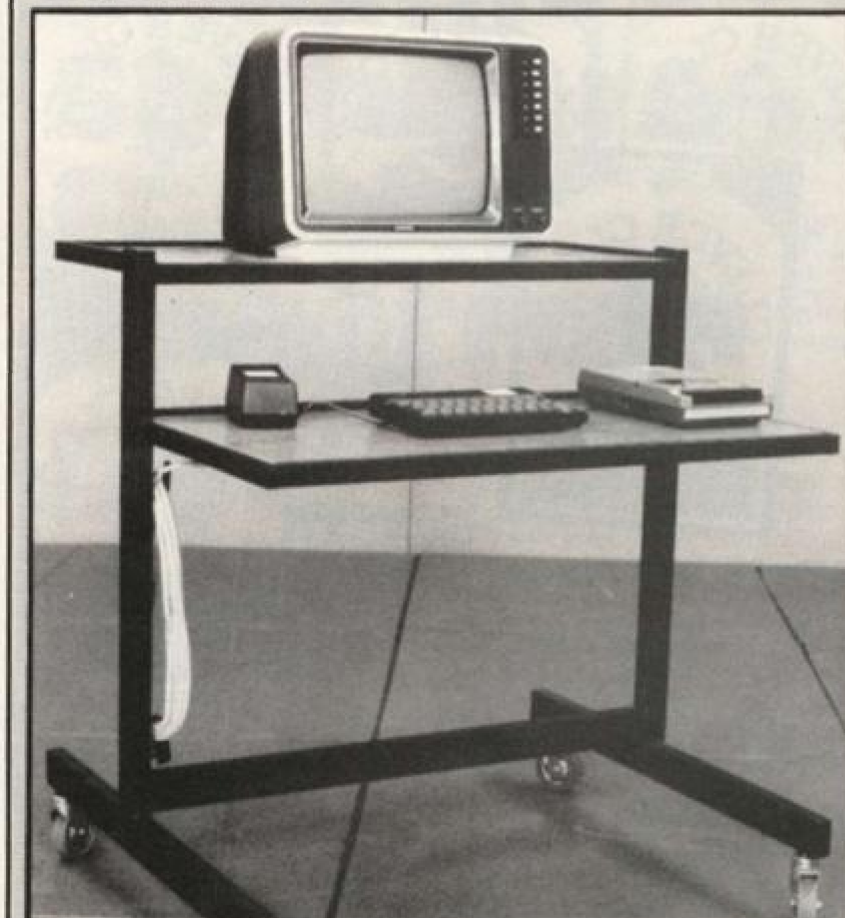
There is one slight problem in that one port is at right angles to the Spectrum and some units won't fit in especially if you have a non standard keyboard.

But as it was designed to go with their Speech unit which is slim and normally lies flat, all it does is raise its position to the more usual upright state.

At £14.95 it is useful, although I can't help feeling that it is more expense that should have been eliminated by the original design of the interfaces.

Now you can use both Micro Speech and the Kempston interface at the same time.

Currah Computer Components Ltd. are at Graythorp ind. Est. Hartlepool, Cleveland.



• A clever idea from Glaston Computers, Yallands Hill, Monkton Heathfield, Taunton, Som. TA2 8NA is PKP's for the ZX81. What are PKP's you ask, well they are little dimples the size of a ZX81 key that are self adhesive and stick on the keyboard, this gives the key a better, more positive feel — hence the name Positive Keyboard Pads.

You can buy a pack of ten for £1.25 to convert a few specific keys for games playing or a pack of forty to convert the whole keyboard for £2.99.

• Sympathies to Autoram computers who had their demonstration stall wrecked by a storm at the Saudicomputer 84 show, they sent some photos but they were all out of focus, a pity as the one described by them of Mr. Dimitry with a "characteristical" expression looked interesting.

• Many people have been contacting Petron Electronics about their sound unit for the ZX81 and Spectrum and they have asked me to apologise to all their customers.

The reason is that they have formed a new limited company which they are calling Newtech Developments, Stockwide Ltd. to cope with increased sales. This means that all cheques to Petron have to be returned and made out to Newtech Developments.

Telepathy

Keep your mind clean, you never know who is tuning in! Chris H. Sinclair of North Humberside helps to test your ability.

Parapsychology is now a growing area of 'respectable' research. Investigations in this field include research into Telepathy and Precognition and to aid such investigations, Dr J.B. Rhine designed a pack of cards known as Zener cards.

A pack consists of 25 cards, each marked with either a cross, a square, wavy lines, a circle, or a star. In telepathy and precognition experiments a card is randomly chosen by a person and the subject under test must guess which card has been chosen. However, it was found that testers tended to pick the cards in a certain sequence. Subjects can subconsciously pick-up these sequences, leading to misleading results. To overcome this, the card sequence is chosen by computer.

This program simulates such an experiment on your Spectrum, drawing the five Zener cards and asking the subject for 100 guesses. The Spectrum then prints the percentage of correct guesses made.

The Program

User defined graphics are used in printing the 'star' and the 'wavy lines' cards. The data for the graphics is held in lines 5000-5240 and lines 50-80 define these graphics. The subroutine starting at line 4000 clears the screen and draws for five Zener cards. Line 190 defines the variables used in the program. Lines 310-410 form the main guess loop. Lines 320-330 defines a random number between 1 and 5. And lines 340-370 get a keypress between 1 and 5. If the guess is the same as the random number generated then 'correct' is increased by one. Line 390 produces short delay (Approx ¼ of a second) before looping to 310 again. This loop is performed 100 times. Lines 510-520 calculates the percentage of correct guesses whilst line 550-590 prints the percentage in large figures, lines 560-570 print the 'tens' part whilst lines 580-590 print the 'units'.

Variables

CORRECT	— No. of correct guesses made.
NO	— Random no. generated.
GUESS	— No. guessed by subject.
TRIES	— No. of times guess loop is executed.
GO	— Controls position on screen of large no. to be printed.
a,b,c,r,d,y,	— For general use.
G$	— Guess input from keyboard.
Y$	— 'Y' or 'N' input.

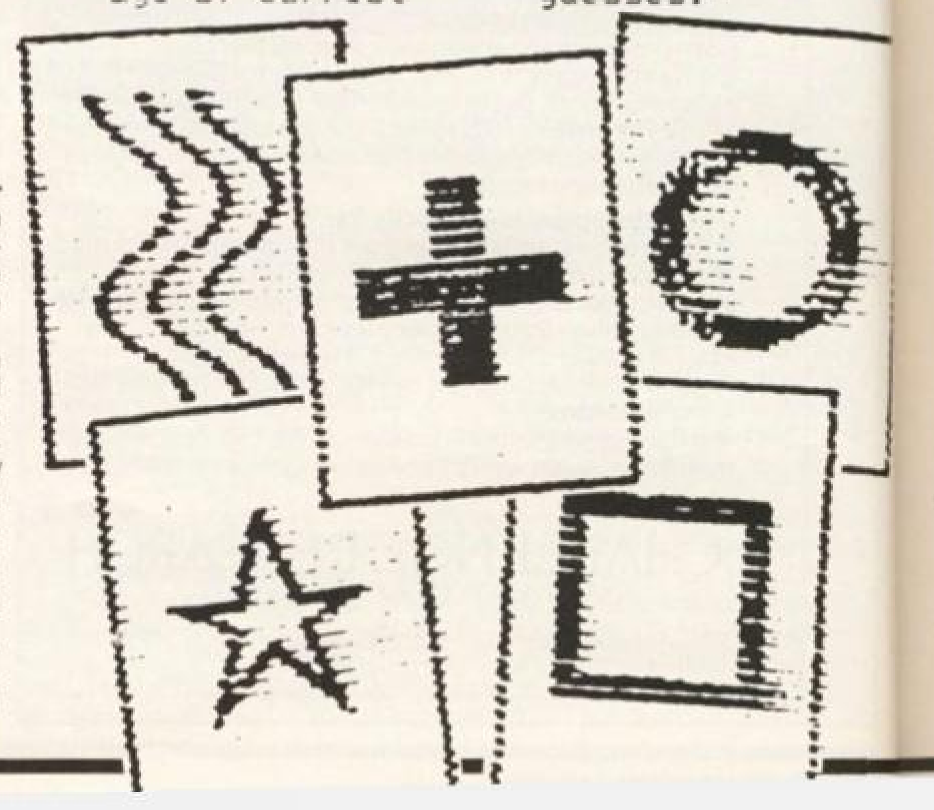
Interpreting Results

It should be realised that what is to follow is an over simplification of what **is** and is not 'significant'. In 5 guesses we may expect to guess 1 correctly by pure chance. Therefore, in 100 guesses we may expect to obtain a 20% score by pure chance. Any score above 20% may, if repeatable, indicate paranormal abilities.

The snag is that statistically speaking, 100 guesses is quite a small sample and so the definition of significance must be altered to account for this. In general it should thus be accepted that scores above around 25% may indicate paranormal abilities.

Finally, those interested in current work in this field could do a lot worse than consult Kit Pedler's book 'Mind Over Matter' published by Thames/Methuen. This book provides an ordinary persons guide to the subject in simple, highly readable language. Above all, it presents a sober, non-sensationalist outlook. A definite antidote for those suffering with an overdose of the Von-Daniken's.

```
   1 REM ***********************
         *Underlined characters*
         *are entered in       *
         *GRAPHICS mode.       *
         ***********************
  40 REM SET UP USER GRAPHICS
  50 RESTORE
  60 FOR A=144 TO 159:
  70 FOR b=0 TO 7: READ c: POKE
USR CHR$ A+b,c: NEXT b
  80 NEXT a
 100 REM INSTRUCTIONS
 110 BORDER 0: PAPER 0: INK 4: C
LS
 120 PAPER 2: INK 6: PRINT AT 0,
10;" TELEPATHY ": PAPER 0: INK 4
 130 PRINT AT 4,0;"You will shor
tly see 5 'Zener   cards' number
ed 1 to 5.I will   randomly sele
ct a card.You must then try to p
redict which card Ihave chosen b
y pressing keys 1  to 5.I will s
ecretly record     whether you a
re correct or      incorrect."
 140 PRINT AT 14,0;"The test con
tinues for 100      guesses afte
r which I will printyour percent
age of correct      guesses."
```



```
 15Ø PAPER 1: INK 7: PRINT AT 21
,3;"PRESS ANY KEY TO CONTINUE"
 16Ø BEEP .Ø5,24
 17Ø PAUSE Ø
 18Ø GO SUB 4ØØØ
 19Ø LET CORRECT=Ø: LET NO=Ø: LE
T GUESS=Ø: LET TRIES=1ØØ: LET CO
=Ø: LET PRINTNUMBER=Ø
 3ØØ REM MAIN GUESS LOOP STARTS
 HERE
 31Ø FOR R=1 TO TRIES
 32Ø LET NO=INT (RND*1Ø)
 33Ø IF NO>5 OR NO<1 THEN  GO TO
 32Ø
 34Ø LET G$=INKEY$: IF INKEY$=""
 THEN  GO TO 34Ø
 35Ø IF G$>"Ø" AND G$<"6" THEN
GO TO 37Ø
 36Ø GO TO 34Ø: REM ERROR IN   I
NPUT
 37Ø LET GUESS=VAL G$: BEEP .Ø2,
12: REM INPUT O.K.
 38Ø IF NO=GUESS THEN  LET CORRE
CT=CORRECT+1
 39Ø FOR D=1 TO 33: NEXT D
 4ØØ NEXT R
 5ØØ REM DISPLAY PERCENTAGE
 51Ø LET PERCENT=(CORRECT/TRIES)
*1ØØ
 52Ø LET PERCENT=INT (PERCENT+.5
)
 53Ø BORDER Ø: PAPER Ø: INK 4: C
LS
 54Ø PRINT " Your score for this
 test is..."
 55Ø OVER 1
 56Ø LET PRINTNUMBER=(PERCENT/1Ø
)-INT (PERCENT/1Ø): LET PRINTNUM
BER=PRINTNUMBER*1Ø
 57Ø LET CO=14: GO SUB 3ØØØ
 58Ø LET PRINTNUMBER=INT (PERCEN
T/1Ø)
 59Ø LET CO=4: GO SUB 3ØØØ
 6ØØ PRINT AT 7,25;"█  █
                         █
                        █
                        █
                       █
                       █
                      █
                     █  █"
 61Ø PAPER 1: INK 7: OVER Ø: PRI
NT AT 21,9;"AGAIN (Y OR N)"
 62Ø LET R$=INKEY$: IF R$="" THE
N  GO TO 62Ø
 63Ø IF R$="Y" OR R$="y" THEN  G
O TO 11Ø
 64Ø IF R$="N"OR R$="n"THEN STOP
 65Ø GO TO 62Ø: REM ERROR IN
 INPUT
3ØØØ REM PRINT PERCENTAGES
3Ø1Ø INK 3: PAPER Ø:
3Ø2Ø GO TO (3Ø3Ø+PRINTNUMBER*1Ø)
3Ø3Ø PRINT AT 7,CO;"██████
                   █    █
                   █    █
                   █    █
                   █    █
                   █    █
                   █    █
                   ██████": R
ETURN
3Ø4Ø PRINT AT 7,CO+7;"█
                      █
                      █
                      █
                      █
                      █
                      █
                      █": RETURN
3Ø5Ø PRINT AT 7,CO;"██████
                        █
                        █
                   ▄▄▄▄▄█
                   █▀▀▀▀▀
                   █
                   █
                   ██████": R
ETURN
3Ø6Ø PRINT AT 7,CO;"██████
                        █
                        █
                   ▄▄▄▄▄█
                   ▀▀▀▀▀█
                        █
                        █
                   ██████": R
ETURN
3Ø7Ø PRINT AT 7,CO;"█    █
                   █    █
                   █    █
                   █▄▄▄▄█
                   ▀▀▀▀▀█
                        █
                        █
                        █": R
ETURN
3Ø8Ø PRINT AT 7,CO;"██████
                   █
                   █
                   █▄▄▄▄▄
                   ▀▀▀▀▀█
                        █
                        █
                   ██████":RETURN
```

```
3090 PRINT AT 7,C0;"████████
                    █
                    █
                    ▀▀▀▀▀▀▀▀
                    █▀▀▀▀▀▀█
                    █      █
                    █      █
                    ████████": R
ETURN
3100 PRINT AT 7,C0;"████████
                           █
                           █
                           █
                           █
                           █
                           █
                           █": R
ETURN
3110 PRINT AT 7,C0;"████████
                    █      █
                    █      █
                    ▀▀▀▀▀▀▀█
                    █▀▀▀▀▀▀█
                    █      █
                    █      █
                    ████████": R
ETURN
3120 PRINT AT 7,C0;"████████
                    █      █
                    █      █
                    ▀▀▀▀▀▀▀█
                    ▀▀▀▀▀▀▀█
                           █
                           █
                    ████████": R
ETURN
4000 REM DRAW ZENER CARDS SUBROU
TINE
4010 REM DRAW CURVES CARD
4020 BORDER 7: PAPER 7: CLS
4030 PAPER 6: FOR a=1 TO 8: PRIN
T AT a,3;"        ": NEXT a
4040 INK 2: PLOT 24,168: DRAW 48
,0: DRAW 0,-65: DRAW -48,0: DRAW
 0,65
4050 INK 0: PRINT AT 2,4;"MMM";A
T 3,5;"PPP";AT 4,5;"OOO";AT 5,4;
"NNN";AT 6,4;"MMM";AT 7,5;"PPP"
4060 BEEP .05,12
4100 REM DRAW STAR CARD
4110 FOR a=1 TO 8: PRINT AT a,13
;"        ": NEXT a
4120 INK 2: PLOT 104,168: DRAW 4
8,0: DRAW 0,-65: DRAW -48,0: DRA
W 0,65
4130 INK 0: PRINT AT 3,15;"KL";A
T 4,14;"GHIJ";AT 5,15;"EF";AT 6,
14;"ABCD"
4140 PLOT 127,152: PLOT 128,152
4150 BEEP .05,12
4200 REM DRAW CIRCLE CARD
4210 FOR a=1 TO 8: PRINT AT a,23
;"        ": NEXT a
4220 INK 2: PLOT 184,168: DRAW 4
8,0: DRAW 0,-65: DRAW -48,0: DRA
W 0,65
4230 INK 0: FOR a=12 TO 16: CIRC
LE 207,136,a: NEXT a
4240 BEEP .05,12
4300 REM DRAW CROSS CARD
4310 FOR a=12 TO 19: PRINT AT a,
8;"        ": NEXT a
4320 INK 2: PLOT 64,80: DRAW 48,
0: DRAW 0,-65: DRAW -48,0: DRAW
0,65
4330 INK 0: PRINT AT 14,10;" █ ";
AT 15,9;"▄▄█▄▄";AT 16,9;"▀▀█▀▀";AT
 17,10;" █ "
4400 REM DRAW SQUARE CARD
4410 FOR a=12 TO 19: PRINT AT a,
18;"        ": NEXT a
4420 INK 2: PLOT 144,80: DRAW 48
,0: DRAW 0,-65: DRAW -48,0: DRAW
 0,65
4430 INK 0: PRINT AT 14,19;"█▀▀▀▀█
";AT 15,19;"█    █";AT 16,19;"█    █
";AT 17,19;"█▄▄▄▄█"
4440 BEEP .05,12
4450 PAPER 7: INK 0: PRINT AT 10
,5;"1";AT 10,15;"2";AT 10,25;"3"
;AT 21,10;"4";AT 21,20;"5"
4460 RETURN
5000 REM DATA FOR STAR GRAPHICS
5010 DATA 1,1,1,3,3,3,2,4
5020 DATA 184,240,224,192,128,0,
0,0
5030 DATA 29,15,7,3,1,0,0,0
5040 DATA 128,128,128,192,192,19
2,64,32
5050 DATA 224,48,96,97,99,199,20
6,220
5060 DATA 7,12,6,134,198,227,115
,59
5070 DATA 0,0,0,0,127,63,14,3
5080 DATA 12,12,24,24,248,240,0,
128
5090 DATA 48,48,24,24,31,15,0,1
5100 DATA 0,0,0,0,254,252,112,19
2
5110 DATA 1,3,3,3,6,6,6,12
5120 DATA 128,192,192,192,96,96,
96,48
5200 REM DATA FOR CURVE GRAPHICS
5210 DATA 96,96,48,48,24,12,6,3
5220 DATA 3,6,12,24,48,48,96,96
5230 DATA 6,6,12,12,24,48,96,192
5240 DATA 192,96,48,24,12,12,6,6
```

Streets of London



High priest of the ZX80, Mike Hyams, calls on his followers to solve this classic adventure.



All regular readers will know that a program from Mike is a rare treat indeed. We always get a spate of letters from users of the other Sinclair machines who convert Mike's programs saying how worthwhile the exercise is. Anyway, conversion is good for the soul!

Unfortunately, in many adventures, by typing in the listing you get to know how to solve it. Mike has come up with a solution to this problem by encoding much of the text — lines 9000+. Because of this you will have to be even more careful than usual when entering this program, trying to debug an encoded program must be a real nightmare!

When the program is run you will find yourself on the platform of High Wycombe British Rail Station, nearly a fate worse than death itself (unless you happen to live in High Wycombe!) You will find that you can only move in certain directions and will have to carry certain objects which you have to find in order to pass some places. Note the use of the word YET may be a clue to whether this place may be passed.

All the commands are entered as one letter inputs — N for north and so on. For a complete list of acceptable commands, type H (for help).

And the best of British. . .

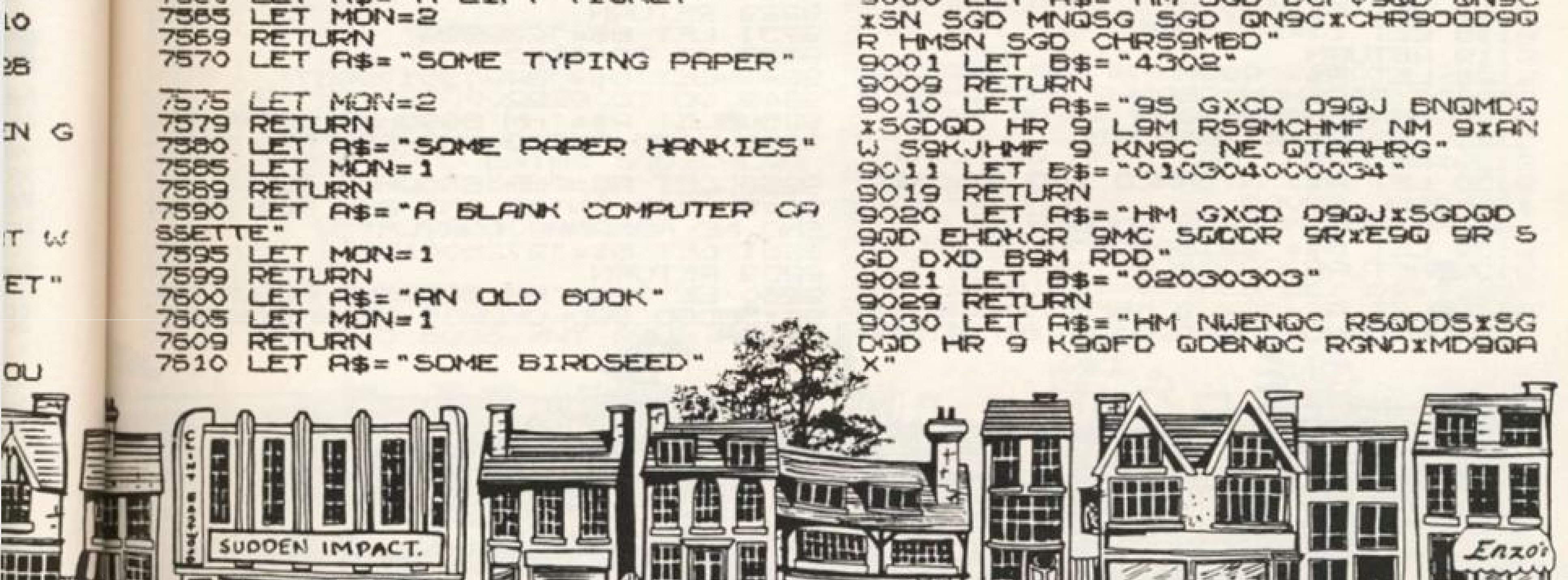
```
  10 GO TO 9900
  40 LET CARRY=0
  50 DIM A(16)
  60 LET SET1=0
  61 LET SET2=0
  75 LET CASH=10
  80 LET T2=8
  85 LET T3=57
  90 LET N=53
 101 LET T3=T3+3
 102 IF T3>59 THEN LET T2=T2+1
 103 IF T3>59 THEN LET T3=T3-60
 104 IF T2>16 AND T3>29 THEN GO
TO 9679
 106 LET IT=0
 107 GO SUB 8990+Nx10
 109 PRINT "YOU ARE ";
 120 LET B=CODE(A$)
 130 IF B=0 THEN GO TO 150
 135 IF B=222 THEN PRINT
 136 IF B=222 THEN GO TO 160

 140 LET B=B+1
 150 PRINT CHR$(B);
 160 LET A$=TL$(A$)
 170 IF NOT A$="" THEN GO TO 120
 180 PRINT
 185 IF B$="STOP" THEN GO TO 986
0
 186 IF B$="AA" THEN GO TO 9885
 188 LET C$=B$
 190 INPUT Z$
 193 LET B$=C$
 195 LET MO=0
 200 IF Z$="N" THEN LET MO=1
 205 IF Z$="S" THEN LET MO=2
 210 IF Z$="E" THEN LET MO=3
 215 IF Z$="W" THEN LET MO=4
 220 IF Z$="U" THEN LET MO=5
 225 IF Z$="D" THEN LET MO=6
 230 IF Z$>"T" OR Z$<"T" THEN GO
TO 234
 231 PRINT ,,,,"TIME ";T2;":";

 232 IF T3<10 THEN PRINT "0";
 233 PRINT T3
 234 IF Z$="M" THEN PRINT ,,,,"M
ONEY £";CASH
 236 IF Z$="B" THEN GO TO 6000
 240 IF Z$="H" THEN GO TO 8000
 243 IF Z$="Q" THEN GO TO 9880
 246 IF Z$="I" THEN GO TO 7000
 250 IF MO=0 THEN GO TO 190
 255 FOR Z=1 TO MO
 260 LET TURN=(CODE(B$)-28)x10
 270 LET B$=TL$(B$)
 280 LET TURN=TURN+CODE(B$)-28
 285 LET B$=TL$(B$)
 290 NEXT Z
 300 IF TURN<1 OR TURN=99 THEN G
O TO 400
 320 LET N=TURN
 325 CLS
 330 GO TO 100
 400 PRINT "YOU CANNOT GO THAT W
AY ";
 410 IF TURN=99 THEN PRINT "YET"
 415 PRINT
 420 GO TO 190
6000 PRINT
6005 IF CARRY>5 THEN PRINT "YOU
```



```
CANNOT CARRY ANY MORE"
6010 IF CARRY>5 THEN GO TO 8120
6015 IF IT=0 THEN GO TO 8120
6020 GO SUB 7500+(ITx10)
6025 IF A(IT)=1 THEN GO TO 6200
6030 IF CASH-MON<0 THEN GO TO 63
00
6050 PRINT "YOU HAVE BOUGHT"
6060 PRINT A$
6070 LET CASH=CASH-MON
6080 LET CARRY=CARRY+1
6090 LET A(IT)=1
6100 PRINT "PRESS N/L TO CONTINU
E"
6105 INPUT Z$
6106 CLS
6110 GO TO 100
6200 PRINT "YOU ALREADY HAVE"
6210 PRINT A$
6220 GO TO 6100
6300 PRINT "YOU DO NOT HAVE ENOU
GH MONEY FOR"
6310 PRINT A$
6320 GO TO 6100
7000 CLS
7010 PRINT "YOU ARE CARRYING..."
7020 PRINT
7023 IF CARRY=0 THEN PRINT "NOTH
ING"
7027 IF CARRY=0 THEN GO TO 8120
7030 FOR Z=1 TO 16
7040 IF A(Z)=0 THEN GO TO 7046
7042 GO SUB 7500+Zx10
7045 PRINT A$
7050 NEXT Z
7060 GO TO 8120
7510 LET A$="A RECORD CALLED xTH
E WALLx"
7515 LET MON=7
7519 RETURN
7520 LET A$="A PAIR OF SHOES"
7525 LET MON=20
7529 RETURN
7530 LET A$="A CINEMA TICKET"
7535 LET MON=3
7539 RETURN
7540 LET A$="AN AUTHORISATION LE
TTER TO        BROADCAST"
7549 RETURN
7550 LET A$="A 194 SWEATSHIRT"
7555 LET MON=5
7559 RETURN
7560 LET A$="A LIFT TICKET"
7565 LET MON=2
7569 RETURN
7570 LET A$="SOME TYPING PAPER"
7575 LET MON=2
7579 RETURN
7580 LET A$="SOME PAPER HANKIES"
7585 LET MON=1
7589 RETURN
7590 LET A$="A BLANK COMPUTER CA
SSETTE"
7595 LET MON=1
7599 RETURN
7600 LET A$="AN OLD BOOK"
7605 LET MON=1
7609 RETURN
7610 LET A$="SOME BIRDSEED"
7615 LET MON=1
7619 RETURN
7620 LET A$="AN AERIAL"
7625 LET MON=15
7629 RETURN
7630 LET A$="A DAY PASS FOR THE
UNDERGROUND"
7635 LET MON=4
7639 RETURN
7640 LET A$="THERMAL UNDERWEAR"
7645 LET MON=20
7649 RETURN
7650 LET A$="A ZX SPECTRUM"
7655 LET MON=100
7659 RETURN
7660 LET A$="A BRITISH RAIL TICK
ET"
7665 LET MON=6
7669 RETURN
8000 CLS
8010 PRINT ,"N...NORTH"
8020 PRINT ,"S...SOUTH"
8030 PRINT ,"E...EAST"
8040 PRINT ,"W...WEST"
8045 PRINT
8050 PRINT ,"U...UP"
8060 PRINT ,"D...DOWN"
8065 PRINT
8070 PRINT ,"T...TIME"
8070 PRINT ,"T...TIME"
8080 PRINT ,"M...MONEY LEFT"
8090 PRINT ,"B...BUY"
8095 PRINT
8100 PRINT ,"I...INVENTORY"
8105 PRINT ,"Q...QUIT"
8110 PRINT
8115 PRINT ,"H...HELP"
8120 PRINT
8130 PRINT "PRESS NEW/LINE TO CO
NTINUE"
8140 INPUT Z$
8145 CLS
8150 GO TO 100
8500 PRINT
8510 PRINT "YOUR MISSION IS COMP
LETED"
8520 PRINT
8530 PRINT "LOOK OUT FOR MORE EX
CITING","ZX-80 GAMES",,,,"ONLY I
N >ZX COMPUTING<"
8999 STOP
9000 LET A$="HM SGD DCFV9QD QN9C
xSN SGD MNQSG SGD QN9CxCHR900D9Q
R HMSN SGD CHRS9MBD"
9001 LET B$="4302"
9009 RETURN
9010 LET A$="95 GXCD 09QJ BNQMDQ
xSGDQD HR 9 L9M RS9MCHMF NM 9xAN
W S9KJHMF 9 KN9C NE QTAAHRG"
9011 LET B$="0100304000034"
9019 RETURN
9020 LET A$="HM GXCD 09QJxSGDQD
9QD EHDKCR 9MC SQDDR 9RxE9Q 9R 5
GD DXD B9M RDD"
9021 LET B$="02030303"
9029 RETURN
9030 LET A$="HM NWENQC RSQDDSxSG
DQD HR 9 K9QFD QDBNQC RGNDxMD9QA
X"
```



```
9031 LET B$="000506020035"
9039 RETURN

9040 LET A$="HM SGD GLU QDBNQC R
SNQD SGD Q9BJR 9QD ETKK NExOHMJ
EKNXCR    +SGD V9KK+"
9041 LET B$="04"
9045 LET IT=1
9049 RETURN
9050 LET A$="HM NWENQC RSQDDS 9S
 9xK9QFD BQNRRQN9CRxSGDQD 9QD RG
ND RGNOR 9KK NUDQxSGD OK9BD"
9051 LET B$="084517040036"
9055 LET IT=2
9059 RETURN
9060 LET A$="HM OHBB9CHKKX BHQBT
RxSGDQD HR 9 ANW NEEHBD RDKKHMFx
SHBJDSR ENQ SGD BHMDL9"
9061 LET B$="450019"
9065 LET IT=3
9069 RETURN
9070 LET A$="HM TOODQ QDFDMS RSQ
DDSxSGD ONRS NEEHBD SNVDQ KNNLRx
9ANUD SGD ATHKCHMFR"

9071 LET B$="09050048"
9079 RETURN
9080 LET A$="NTSRHCD SGD K9MFG9L
xGNSDKxSGDQD HR'9M NKC ATHKCHMFx
NOONRHSD XNT ATS HS KNNJR UDQXxA
NQHMF"
9081 LET B$="100814"
9089 RETURN
9090 LET A$="NTSRHCD SGD AAB GPx
E9LNTR QDNOKD JDDO O9RRHMFxSGQNT
FG SGD CNNQR 9MC HFMNQHMFxXNT"
9091 LET B$="000911"
9099 RETURN
9100 LET A$="HM SGD QDBDOSHNM NE
xSGD AABxQ9CHN O FDMSKX CQHESR E
QNL 9xRODD9JDQ"
9101 LET B$="00000010"
9102 IF SET2=0 THEN LET A$="HM S
GD QDBDOSHNM NExSGD AABx9 RL9QSK
X CQDRRDC L9M G9MCR XNT 9M DMUDK
NOD"

9103 IF SET2=0 THEN LET CARRY=CA
RRY+1
9104 LET A(4)=1
9105 LET SET2=1
9109 RETURN
9110 LET A$="HM SGD B90H59K Q9CH
NxENXDQxSGDQD HR 9 RGNO RDKKHMF
O83xRVD9SRGHQSR"
9111 LET B$="0013"
9115 LET IT=5
9119 RETURN
9120 LET A$="RS9MCHMF AX V9QQDMx
RSQDDS RS9SHNMx9BQNRR SGD QN9C H
R 9M DMNQLNTRxFQDDM SNVDQ AKNBJ"
9121 LET B$="12230022"
9129 RETURN
9130 LET A$="NTSRHCD SGD AQHSHRG
xSDKDBNL SNVDQ
 HS HR UDQX VHMCX GDQD"
9131 LET B$="22150009"
9139 RETURN

9140 LET A$="AX 9 KHES HM SGDxAS
 SNVDQxODNOKD RS9QD 9S XNT RTRDH
BHNTRKX"
9141 LET B$="1400000042"
9142 IF A(6)=0 THEN LET B$="140
000099"
9145 LET IT=6
9149 RETURN
9150 LET A$="HM QXL9MR SGD RS9S
NMDQRSGDQD HR SXOHMF O9ODQ ENQ 
9KD"
9151 LET B$="0016"
9155 LET IT=7
9159 RETURN
9160 LET A$="HM NWENQC RSQDDSxS
 SGD D9RS SGD BDMSQD ONHMSxSNVDG
 RS9MCR GHFG 9ANUD SGDxRGNOR"
9161 LET B$="16002506"
9169 RETURN
9170 LET A$="HM SGD ENXDQ NE SGD
xDLOHQD BHMDL9xSGDQD HR 9 EHKL 

9170 LET A$="HM SGD ENXDQ NE SGD
xDLOHQD BHMDL9xSGDQD HR 9 EHKL 
ANTS SN RS9QSxSGD RVDDS BNTMSDQ
HR RDKKHMFxG9MJHDR"
9171 LET B$="0019000040"
9172 IF A(3)=0 THEN LET B$="001
000099"
9175 LET IT=8
9179 RETURN
9180 LET A$="HM KDHBDRSDQ RPT9QD
xSGDQD 9QD BHMDL9R DUDQXVGDQD"
9181 LET B$="18262607"
9189 RETURN
9190 LET A$="HM SGD YW BNLOTSHMF
xNEEHBDRxSGD RDBQDS9QX SDKKR XNT
 SN S9JD SGD KHES TO"
9191 LET B$="0000270050"
9192 IF A(4)=0 OR A(12)=0 OR A(1
5)=0 OR A(14)=0 THEN LET B$="000
0270099"
9199 RETURN
9200 LET A$="HM K9RJXR SGD GHEH 
RGNO SGDQD 9QD AK9MJ BNLOTSDQx59
ODR ENQ R9KD"
9201 LET B$="000023"
9205 LET IT=9
9209 RETURN
9210 LET A$="NTSRHCD FS ONQSK9MC
 RSxRS9SHNMxSGDQD HR 9 O9QJ SN S
GD MNQSG"
9211 LET B$="461413"
9219 RETURN
9220 LET B$="13240021"
9222 LET A$="HM SNSSDMG9L BNTQS 
QN9C"
9229 RETURN
9231 LET B$="232529"
9239 GO TO 9222
9241 LET B$="242730170037"
9249 GO TO 9222
9250 LET A$="HM BG9QHMF BQNRR QN
9CxSN SGD RNTSG XNT B9M RDD SGDx

9250 LET A$="HM BG9QHMF BQNRR QN
9CxSN SGD RNTSG XNT B9M RDD SGDx
SNO NE MDKRNMR BNKTLM"
9251 LET B$="27280019"
9259 RETURN
9260 LET A$="HM BG9QHMF BQNRR QN
9CxSGDQD 9QD OKDMSX NE NKCxANNJR
GNOR"
```



```
9261 LET B$="25260020"
9265 LET IT=10
9269 RETURN
9270 LET A$="HM SQ9E9KF9Q RPT9QD
xSGDQD 9QD SGNTR9MCR NE OHFDNMRx
EKXHMF 9QNTMCxSGDQD HR 9M NKC L9
M RDKKHMFxAHQCRDDC"
9271 LET B$="26003219"
9275 LET IT=11
9279 RETURN
9280 LET A$="NM SGD FQNTMC EKNNQ
 NExSGD KHNM GHEH RSNQDxXNT B9M
ATX 9M 9DQH9K 95xUDQX BNLODSHSHU

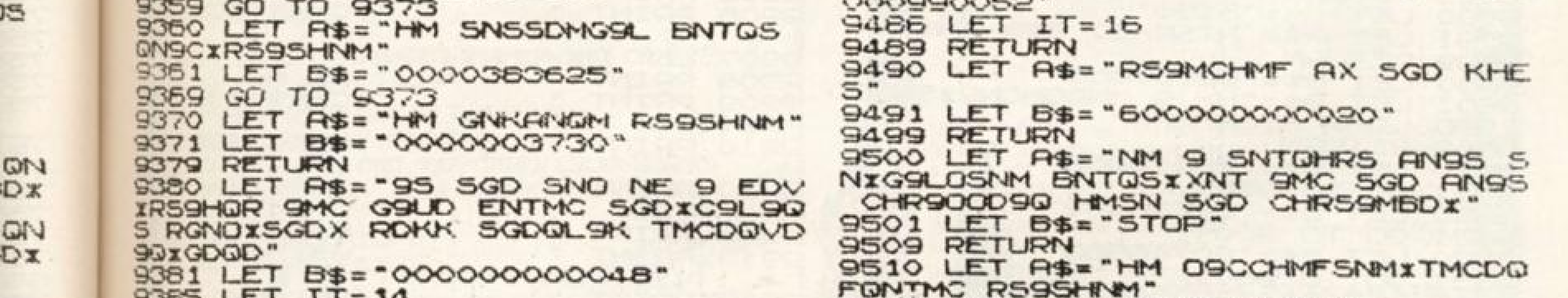
9280 LET A$="NM SGD FQNTMC EKNNQ
 NExSGD KHNM GHEH RSNQDxXNT B9M
ATX 9M 9DQH9K 95xUDQX BNLODSHSHU
D OQHBDR GDQD"
9281 LET B$="0000002441"
9285 LET IT=12
9289 RETURN
9290 LET A$="HM SGD GNKANQM"
9291 LET B$="000000250038"
9299 RETURN
9300 LET A$="HM BNUDMS F9QCDMxSG
DQD 9QD SD9 RGNOR CN55DCx9QNTMC"
9301 LET B$="00320047"
9309 RETURN
9310 LET A$="HM SGD RSQ9MCxXNT B
9M RDD O9QS NE SQ9E9KFDQxRPT9QD
EQNL GDQD"
9311 LET B$="31330028"
9319 RETURN
9320 LET A$="NM SGD DLA9MJLDMSx9
 BNTOKD NE SNTQHRS AN9SRxFDMSKX

9320 LET A$="NM SGD DLA9MJLDMSx9
 BNTOKD NE SNTQHRS AN9SRxFDMSKX
R9HK O9RS"
9321 LET B$="3251"
9329 RETURN
9330 LET A$="HM L9QAKD 9QBGxTMCD
QFQNTMC RS9SHNMxSGD SQ9HMR SQ9UD
K D9RS 9MC VDRS EQ@L GDQD"
9331 LET B$="0000354902"
9339 GO TO 9373
9340 LET A$="HM ANMC RSQDDS RS95
HNM"
9341 LET B$="0000363404"
9349 GO TO 9373
9350 LET A$="HM NWENQC BHQBTR RS
95HNM"
9351 LET B$="0000373506"
9359 GO TO 9373
9360 LET A$="HM SNSSDMG9L BNTQS
QN9CxRS95HNM"
9361 LET B$="0000383625"
9369 GO TO 9373
9370 LET A$="HM GNKANQM RS95HNM"
9371 LET B$="0000003730"
9379 RETURN
9380 LET A$="95 SGD SNO NE 9 EDV
xRS9HQR 9MC G9UD ENTMC SGDxC9L9Q
5 RGNDxSGDX RDKK SGDQL9K TMCDQVD
9QxGDQD"
9381 LET B$="00000000048"
9385 LET IT=14
9389 RETURN
9390 LET A$="ITRS NM XNTQ V9X NT
S NE SGD BHMDL9 9ESDQ V9SBGHMF =
DS)xH B9M RDD SG95 XNT G9UD ADDM
xBQXHMF 9R XNT G9UD MN G9MJHDRxK
DES"
9391 LET B$="00000000000018"
9392 LET A(3)=0
9393 LET CARRY=CARRY-1
9399 RETURN
9400 LET A$="HM SGD BNLOTSDQxCDO

9400 LET A$="HM SGD BNLOTSDQxCDO
9QSLDMS NE KHNM GNTRDxSGDX G9UD
RODBSQTLR ENQ R9KD"
9401 LET B$="000000000029"
9405 LET IT=15
9409 RETURN
9410 LET A$="NM SGD NARDQU9SHNM
EKNNQNE SGD SNVDQxUDQX RBDMHB AT
S SN59KKXxONHMSKDRR HL 9EQ9HC"
9411 LET B$="00000000015"
9419 RETURN
9420 LET A$="HM SGD DCFV9QD QN9C
xSGD QN9C RSHKK CHR9OOD9QR HMSNx
SGD MNQSG"
9421 LET B$="4401"
9429 RETURN
9430 LET A$="HM RS 9KA9MRxXNT V9
KJDC SNN E9Q TO SGDxDCFV9QD QN9C
x"
9431 LET B$="STOP"
9439 RETURN

9440 LET A$="HM QDFDMS RSQDDS"
9441 LET B$="0607"
9449 RETURN
9450 LET A$="HM QDFDMSR O9QJxSGD
QD 9QD OQDSSX EKNVDQR CNSSDC 9QN
TMC"
9451 LET B$="46224646"
9459 RETURN
9460 LET A$="HM XD NKCD SD9 RGNO
ODx9ESDQ 9 QDK9WHMF BTO NE SD9 X
NT G9UD SN BNMSHMTD XNTQ RD9QBG"
9461 LET B$="000051"
9462 LET T2=T2+1
9469 RETURN
9470 LET A$="HM 9 QDBDOSHNM ENXD
Qx9 RDBQDS9QX RLHKDR 95 XNT"
9471 LET B$="0000060039"
9479 RETURN
9480 LET A$="HM O9CCHMFSNM AQ RS
95HNM"
9481 LET B$="0000000530052"

9482 IF A(16)=0 THEN LET B$="000
000990052"
9486 LET IT=16
9489 RETURN
9490 LET A$="RS9MCHMF AX SGD KHE
5"
9491 LET B$="60000000000020"
9499 RETURN
9500 LET A$="NM 9 SNTQHRS AN95 S
NxG9LOSNM BNTQSxXNT 9MC SGD AN95
 CHR9OOD9Q HMSN SGD CHR59MBDx"
9501 LET B$="STOP"
9509 RETURN
9510 LET A$="HM O9CCHMFSNMxTMCDQ
FQNTMC RS95HNM"
9511 LET B$="0034000049"
9512 IF A(13)=0 THEN LET B$="009
9000049"
9513 IF A(16)=0 THEN GO TO 9516
```



```
9514 LET A(16)=0
9515 LET CARRY=CARRY-1
9516 LET IT=13
9519 RETURN
9520 LET A$="HM GHFG VXBNLAD AGH
SHRG Q9HK R595HNMxKNMCNM 5Q9HMR
FN D9R5 EQNL GDQD"
9521 LET B$="00004954"
9522 IF A(16)=0 THEN LET B$="000
099"
9525 LET IT=16
9529 RETURN
9530 LET A$="HM NWENQC R59SHNMxS
GDQD 9QD TMHUDQRHSX ATHKCHMFRx9K
K 9QNTMC"
9531 LET B$="00555356"
9532 IF A(16)=0 THEN LET B$="005
59958"
9535 LET IT=16
9539 RETURN
9540 LET A$="AX SGD ATR RSNO HMx
NWENQC BHSX BDMSQD"
9541 LET B$="54575658"

9542█IF A(16)=0 THEN GO TO 9545
9543 LET A(16)=0
9544 LET CARRY=CARRY-1
9549 RETURN
9550 LET A$="NTSRHCD NWENQC TSCx
ENNSA9KK FQNTMC"
9551 LET B$="00000055"
9559 RETURN
9560 LET A$="95 SGD SDLOKD E9QLx
BNTMSQX BKTA ITRS NTSRHCDxNWENQC
xXNT RODMC SGD QDRS NE SGD C9XxV
95BGHMF AN9SR NM SGD QHUDQx"
9561 LET B$="STOP"
9569 RETURN
9570 LET A$="RS9MCHMF AX 9 K9QFD
 NKC BGTQBG"
9571 LET B$="005955"
9572 IF A(16)=0 THEN GO TO 9575
9573 LET A(16)=0
9574 LET CARRY=CARRY-1
9579 RETURN

9580 LET A$="HM NWENQC BHSX A9MJ
xXNT G9UD VHSGCQ9VM RNLD LNMDX"
9581 LET B$="58"
9584 IF SET1=1 THEN GO TO 9587
9585 LET CASH=CASH+150
9586 LET SET1=1
9589 RETURN
9590 LET B$="6150"
9591 LET A$="NTSRHCD SGD KHES"
9599 RETURN
9600 LET B$="62607562"
9601 LET A$="HM 9 BNQQHCNQxSGDQD
 9QD L9MX STQMHMFR"
9609 RETURN
9610 LET B$="63510082"
9611 GO TO 9601
9620 LET B$="60626462"
9621 GO TO 9601
9630 LET B$="00746563"
9631 GO TO 9601
9640 LET B$="73006664"

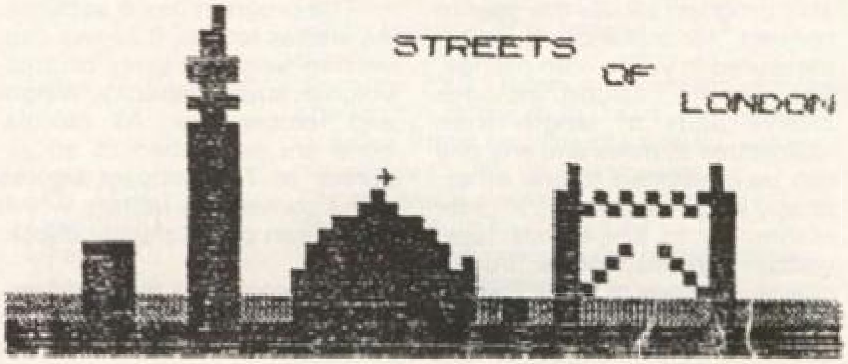
9641 GO TO 9601
9650 LET B$="67006065"
9651 GO TO 9601
9660 LET B$="666662"
9661 GO TO 9601
9670 LET B$="69678261"
9671 GO TO 9601
9680 LET B$="82680070"
9681 GO TO 9601
9690 LET B$="00816971"
9691 GO TO 9601
9700 LET B$="60607072"
9701 GO TO 9601
9710 LET B$="800071"
9711 GO TO 9601
9720 LET B$="00650060"
9721 GO TO 9601
9730 LET B$="640075"
9731 GO TO 9601
9740 LET B$="00006074"
9741 GO TO 9601
9750 LET B$="00787761"
9751 GO TO 9601
9760 LET B$="00000076"
9761 GO TO 9601
9770 LET B$="760079"
9771 GO TO 9601
9780 LET B$="00600078"
9781 GO TO 9601
9790 LET A$="GDQD 95 K9R5xxBNMFQ
95TK95HNMRxxXNT G9UD RDS TO 9 RH
MBK9HQxAQN9CB9RSHMF R595HNM NM S
GDxQNNE"
9791 LET B$="AA"
9799 RETURN
9800 LET B$="70006860"
9801 GO TO 9601
9810 LET A$="E9KKHMF EQ&L 9M NTS
RHCD VHMCNV SN SGD FQNTMC ADKNVx
XNT RODMC SGD MDWS 1 VDDJR HMxSG
D LHCCKDRDW GNROH59K"
9811 LET B$="STOP"

9819 RETURN
9879 PRINT "IT IS 5:30 PM",,,"YO
U ARE TOO LATE"
9880 PRINT "YOUR MISSION HAS FAI
LED"
9881 STOP
9885 GO TO 8500
9899 STOP
9900 CLS
9901 PRINT ,"STREETS"
9902 PRINT ,,"OF"
9903 PRINT ,,"    LONDON"
9904 PRINT
9905 PRINT "              █"
9906 PRINT "              ";CHR$(130);"
█"
9907 LET A$=CHR$(128)
9908 PRINT "              ";A$;A$
9909 PRINT "              ";CHR$(132);C
HR$(133)
9910 PRINT "              ";CHR$(132);C

9910 PRINT "              ";CHR$(132);C
HR$(133)
9911 PRINT "              ";A$;A$
9912 PRINT "              ";A$;A$;"
 +       ";CHR$(130),"    █"
9913 PRINT "              ";A$;A$;"
▪";CHR$(133),"       ";A$;"▀▄▀▄▀▄"; 
A$
9914 PRINT "              ";A$;A$;"
;CHR$(132);A$;A$;A$;"▪      ";A$;"
     ";A$
```



```
9915 PRINT "   ";A$;A$;"   ";A$;A
$;"   ";CHR$(130);A$;A$;A$;A$;A$;
"▖  ";A$;"▗▘ ";CHR$(136);"  ";A$
9916 PRINT "   ";A$;A$;"   ";A$;A
$;"   ";A$;A$;A$;A$;A$;A$;A$;"
";A$;"▗▘  ";CHR$(136);A$
9917 PRINT "▒▒▒▒";A$;A$;"▒▒▒";A$;A
$;"▒▒";A$;A$;A$;A$;A$;A$;A$;"▒";
A$;"▒";A$;"▒▒▒▒▒▒▒";A$;"▒▒▒▒▒"
9919▒FOR B=1 TO 32
9920▒PRINT A$;A$;A$;A$;A$;
9921 NEXT B
9923 PRINT A$;A$;A$;"PRESS NEW-L
INE TO START";
9924 FOR A=1 TO 19
9925 PRINT A$;A$;
9926 NEXT A
9930 INPUT Z$
9935 CLS
9940 PRINT ,"STREETS OF LONDON"
9945 PRINT ,"▀▀▀▀▀▀▀ ▀▀ ▀▀▀▀▀▀"
9947 PRINT
9949 PRINT "THIS IS AN ADVENTURE
 SET IN THE STREETS OF LONDON."
9950 PRINT "YOU HAVE A MISSION T
O COMPLETE   IN ONE DAY. THE TIME
 IS 9:00 AM AND YOU MUST BE FINI
SHED BY      5:30 PM."
9955 PRINT "UNFORTUNATELY YOU CA
NNOT RECALL WHAT IT IS THAT YOU
ARE SUPPOSEDTO BE DOING."
9960 PRINT "YOU MUST WORK FAST T
O DISCOVER   THE SECRET, BUT DO N
OT GET       DISTRACTED BY UNIMPO
RTANT EVENTSOR YOU WILL BE TOO L
ATE."
9965 PRINT
9966 PRINT "YOU START YOUR DAY I
N",,"HIGH WYCOMBE.   GOOD LUCK"
9970 PRINT
9975 REM PRINT "PRESS > H <,DURI
NG PLAY, FOR A  LIST OF POSSIBLE
 COMMANDS."
9977 PRINT ,"PRESS N/L TO START"
9980 INPUT Z$
9981 CLS
9982 GO TO 12
9989 STOP
9990 REM
```



Conversions

A program with many uses from kitchen to work shop from Jonathan Court of Poole.

This program allows the user to convert, for instance, a length measured in yards, into metres. The 'length' option includes **twelve units of length from** millimetres to miles and any one can be converted to any other, so any length expressed in units of mm, cm, m, km, inches, feet, yards, fathoms, poles (rods), chains, furlongs or miles, can be easily converted into the units of your choice. This section of the program is very useful for anyone who prefers miles to km, inches to cm, or vice versa.

The program has 6 sections. As well as length, it allows conversion between units of area, volume, liquid, capacity, weight and temperature. All calculations are performed to an accuracy of **7 significant figures** using conversion factors which have been painstakingly checked.

The program is menu-driven, and fig.1 shows a screen dump of the menu. The user enters the number he requires, and one of six lists results (see fig.2). You are then required to enter the number of the units that you wish to change.

Let us suppose that you wish **to convert 10cm into inches. You** enter "2" in reply to the prompt "FROM?", and "5" in reply to the prompt "TO?". The screen will then appear as shown in fig.3. You are required to enter the number of cm, 10, in our case, and this is immediately converted into inches and displayed to seven significant figures (fig.4). Pressing "L" will return the user to the list of lengths and the "FROM?" prompt, and pressing "M" will return you to the menu. This procedure is the same for the other sections.

The temperature section only contains three units of temperature: degrees Centigrade, Fahrenheit and Kelvin. This is almost the same as before, the answer being given to two decimal places, since seven-figure accuracy seemed inappropriate. An exampl temperature is shown in fig.5.

Any of the six sections, alon with the subroutine at the end o the program (not necessary fo temperature) could be used as a program on its own, and m routines for giving significan figures and decimal places, both of which are explained later could easily be used in other programs.

The method I have devise for converting between units is a simple one, and will not produce answers such as 11.99998 when the answer ought to be 12 or whatever. This is achieved by using only one complicated conversion factor, the others being simple powers of ten of this number (for metric units) or simple whole number multiples or reciprocols for imperial units. Hence the answers will always be exact and correct to at least seven significant figures, should this be required.

```
C O N V E R S I O N

1. LENGTH
2. AREA
3. VOLUME
4. LIQUID CAPACITY
5. WEIGHT
6. TEMPERATURE

INPUT REQUIRED NUMBER
```

Fig.1. Screen dump of menu.

Program notes

Line 8	This provides a neat method of clearing a line whenever necessary without having to type in a line of spaces.
Lines 10-120	Here, the menu is printed on the screen.
Lines 130-150	This INPUTs the number, checks it for mistakes, and calls the relevant section of the program.
Line 1000	This is the start of the section dealing with length conversion.

```
L E N G T H

1.   MILLIMETRES
2.   CENTIMETRES
3.   METRES
4.   KILOMETRES
5.   INCHES
6.   FEET
7.   YARDS
8.   FATHOMS
9.   POLES(RODS)
10.  CHAINS
11.  FURLONGS
12.  MILES

FROM ?
```

Fig.2(a). Length conversion options.

```
A R E A

1.   SQUARE MM
2.   SQUARE CM
3.   SQUARE METRES
4.   SQUARE KM
5.   CENTIARES
6.   ARES
7.   HECTARES
8.   SQUARE INCHES
9.   SQUARE FEET
10.  SQUARE YARDS
11.  PERCHES
12.  SQUARE CHAINS
13.  ROODS
14.  ACRES
15.  SQUARE MILES
FROM ?
```

Fig.2(b). Area conversion table.

Line 1010 — This is the conversion figure used to convert between metric and imperial lengths. It is actually the number of yards in a metre.

Line 1050 — An array is set up to hold the twelve units of length.

Line 1060 — This array holds the twelve conversion factors. In this array, every unit of length is expressed in terms of yards. This involves numbers, e.g. 1760 for a mile, for imperial lengths, and the conversion factor from line 1050 multiplied by a power of ten for metric lengths.

Lines 1070-1300 — The arrays are then filled with the appropriate entries.

Lines 1310-1360 — The list of lengths is printed on the screen. In line 1350, the part between the brackets prints a space before the number if it is less than ten. This makes for a neater screen layout.

Line 1370 — This sets the line on the screen at which the prompts will start.

Line 1380 — This is the number of lengths. The subroutine being called by line 1390 needs to know this so that it can check for invalid entries.

Line 7000 — (This is described here for convenience) This line prints the prompt at the line specified by 1370. This position varies according to the length of the list of units in the section.

Lines 7010-7030 — The unit to be converted from is entered, checked for invalidity, and displayed next to the word "FROM", over the question mark.

Lines 7040-7070 — The unit to be converted to is chosen, checked and displayed.

Lines 7080-7100 — The actual length is entered and checked in case it is zero. Entering zero would be pointless unless it was a temperature, and the temperature section does not use this subroutine.

Line 7110 — The three (or possibly four) lines just used are cleared using S$ (line 8).

Lines 7120-7130 — The original length is displayed along with "IS EQUAL TO".

Line 7140 — This is a very important line. The length (X) is multiplied by the necessary number to give it in terms of its new units. This number is obtained by dividing the value in B (the lengths expressed in yards) of the final units by the value for the original units. The outcome, Y, is the answer. It may be acceptable as it is, but it may be recurring or expressed to a number of decimal places greater than the accuracy of the conversion figures.

Lines 7150-7160 — This routine determines the final number of significant figures. The two (6 − Z)s give seven significant figures. Any number, x, can be used in place of 6, and the number Y will be reduced (if ncessary) to x + 1 significant figures.

Lines 7170-7180 — Since the ZX81 prints numbers smaller than one as .xyz rather than 0.xyz, I have included these two lines to add a 0 where necessary to tidy up the answer.

Line 7190 — The answer is printed.

Line 7200 — We now return to line 1390

Lines 1400-1410 — Instructions to proceed are printed at the bottom of the screen.

Line 1420 — The program waits until a key is pressed.

Lines 1430 — In order to check validity, a variable is assigned. This is probably not necessary, but if the validity checking is more than a couple of lines, it is possible to take one's finger off the appropriate key before the computer has finished.

Lines 1440-1460 — The user is returned to the list of lengths or to the menu. 1460 returns to 1410 if an invalid key has been pressed by mistake.

Line 2000 — This is the beginning of the section dealing with area conversion.

```
V O L U M E

1.   CUBIC MILLIMETRES
2.   CUBIC CENTIMETRES
3.   CUBIC DECIMETRES
4.   CUBIC METRES (STERES)
5.   CUBIC INCHES
6.   CUBIC FEET
7.   CUBIC YARDS

FROM ?
```

Fig.2(c). Volume conversion table.

```
W E I G H T

1.    MILLIGRAMS
2.    GRAMS
3.    KILOGRAMS
4.    TONNES
5.    DRAMS
6.    OUNCES
7.    POUNDS
8.    STONES
9.    QUARTERS
10.   HUNDREDWEIGHT
11.   TONS

FROM ?
```

Fig.2(e). Weight conversion table.

```
L I Q U I D   C A P A C I T Y

1.   MILLILITRES
2.   CENTILITRES
3.   LITRES
4.   FLUID OUNCES
5.   GILLS
6.   PINTS
7.   QUARTS
8.   GALLONS

FROM ?
```

Fig.2(d). Capacity conversion table.

```
T E M P E R A T U R E

1. DEGREES CENTIGRADE (CELSIUS)
2. DEGREES FAHRENHEIT
3. KELVIN

FROM ?
```

Fig.2(f). Temperature conversion table.

The structure of this section is exactly the same as the length section. The number in line 2010 is the number of square chains in 1000 square metres.

Line 3000 This is the beginning of the volume section. This is also the same as the length section, and the number in line 3010 is the number of cubic inches in 100 cubic centimetres.

Line 4000 This is the liquid capacity section. The number in line 4010 is the number of pints in a litre.

Line 5000 This is the section dealing with weight. The number in line 5010 is the number of pounds in a kilogram. I have used the English spellings of the metric weights rather than the French spellings which seem to have fallen into disuse.

Line 6000 This is the temperature section. It is not the same as the others because simple scalar conversion factors cannot be used since none of the scales share the same zero point.

Lines 6010-6040 The names of the scales are put into the array A$.

Lines 6050-6090 The list is printed on the screen.

Lines 6100-6240 This is similar to the other sections, and allows the user to choose from and to which he is converting, and to input his temperature.

Lines 6250-6260 The temperature is converted into degress Centigrade if it was originally in Kelvin or degrees Fahrenheit.

Lines 6270-6280 The temperature, now in degrees Centigrade, is here converted to the units of your choice if they are not degrees C.

Line 6290 This line reduces the answer to two decimal places. The two 100s will give two decimal places. 1000s would give three, and so on.

Line 6300 The answer is printed by this line.

Lines 6310-6380 This sends the user wherever he now wishes to go, as in previous sections.

Lines 7000-7200 This is the subroutine used in sections one to five and described with section one.

Lines 8000-8010 This SAVEs the program so that it will RUN automatically on loading. The program should be SAVEd by: RUN 8000

Spectrum compatibility

To the best of my knowledge, this program will run if typed into any Spectrum. It should be noted that the * * appearing twice in line 7160 is not two *s, but the ZX81 version of the symbol, found on the 'H' key.

```
L E N G T H

 1.     MILLIMETRES
 2.     CENTIMETRES
 3.     METRES
 4.     KILOMETRES
 5.     INCHES
 6.     FEET
 7.     YARDS
 8.     FATHOMS
 9.     POLES(RODS)
10.     CHAINS
11.     FURLONGS
12.     MILES

FROM CENTIMETRES
TO INCHES
INPUT NO. OF CENTIMETRES
```

Fig.3. Length conversion screen.

```
   2 LET L$=""
   3 FOR I=1 TO 30
   4 LET L$=L$+CHR$ 131
   5 NEXT I
   8 DIM S$(32)
  10 CLS
  20 PRINT "******************
************"
  30 PRINT "*";TAB 31;"*"
  40 PRINT "*        C O N V E R S
I O N       *"
  50 PRINT "*";TAB 31;"*"
  60 PRINT "******************
************"
  70 PRINT AT 7,7;"1. LENGTH"
  80 PRINT AT 9,7;"2. AREA"
  90 PRINT AT 11,7;"3. VOLUME"
 100 PRINT AT 13,7;"4. LIQUID CA
PACITY"
 110 PRINT AT 15,7;"5. WEIGHT"
 115 PRINT AT 17,7;"6. TEMPERATU
RE"
 119 REM *USE INVERSE PRINT IN *
       *NEXTLINE               *
 120 PRINT AT 21,0;"     input r
equired number      "
 130 INPUT A
 140 IF A<1 OR A>6 OR A<>INT A T
HEN GOTO 130
 150 GOTO 1000*A
1000 REM *** length ***
```

```
L E N G T H

 1.     MILLIMETRES
 2.     CENTIMETRES
 3.     METRES
 4.     KILOMETRES
 5.     INCHES
 6.     FEET
 7.     YARDS
 8.     FATHOMS
 9.     POLES(RODS)
10.     CHAINS
11.     FURLONGS
12.     MILES

  10 CENTIMETRES
    IS EQUAL TO
  3.937008 INCHES
HIT   TO CONVERT ANOTHER LENGTH
       HIT M FOR MENU
```

Fig.4. Example output from Length option.

```
T E M P E R A T U R E

1. DEGREES CENTIGRADE (CELSIUS)
2. DEGREES FAHRENHEIT
3. KELVIN


FROM ?
```

Fig.5(a) Temperature options available.

```
1Ø1Ø LET Y=1.Ø936133
1Ø5Ø DIM A$(12,11)
1Ø6Ø DIM B(12)
1Ø7Ø LET A$(1)="MILLIMETRES"
1Ø8Ø LET B(1)=Y/1ØØØ
1Ø9Ø LET A$(2)="CENTIMETRES"
11ØØ LET B(2)=Y/1ØØ
111Ø LET A$(3)="METRES"
112Ø LET B(3)=Y
113Ø LET A$(4)="KILOMETRES"
114Ø LET B(4)=Y*1ØØØ
115Ø LET A$(5)="INCHES"
116Ø LET B(5)=1/36
117Ø LET A$(6)="FEET"
118Ø LET B(6)=1/3
119Ø LET A$(7)="YARDS"
12ØØ LET B(7)=1
121Ø LET A$(8)="FATHOMS"
122Ø LET B(8)=2
123Ø LET A$(9)="POLES(RODS)"
124Ø LET B(9)=5.5
125Ø LET A$(1Ø)="CHAINS"
126Ø LET B(1Ø)=22
127Ø LET A$(11)="FURLONGS"
128Ø LET B(11)=22Ø
129Ø LET A$(12)="MILES"
13ØØ LET B(12)=176Ø
131Ø CLS
132Ø PRINT TAB 9;"L E N G T H"
133Ø PRINT TAB 8;L$( TO 13)
134Ø FOR N=1 TO 12
```

```
T E M P E R A T U R E

1. DEGREES CENTIGRADE (CELSIUS)
2. DEGREES FAHRENHEIT
3. KELVIN

FROM DEGREES FAHRENHEIT

TO DEGREES CENTIGRADE (CELSIUS)

INPUT TEMPERATURE IN
DEGREES FAHRENHEIT
```

Fig.5(b) Conversion units selected.

```
T E M P E R A T U R E

1. DEGREES CENTIGRADE (CELSIUS)
2. DEGREES FAHRENHEIT
3. KELVIN

98.4
DEGREES FAHRENHEIT

   IS EQUAL TO

36.89
DEGREES CENTIGRADE (CELSIUS)

HIT T TO CONVERT ANOTHER
     TEMPERATURE
   HIT M FOR MENU
```

Fig.5(c) Example output from Temperature option.

```
135Ø PRINT AT N+3,Ø+(1 AND N<=9)
;N;".     ";A$(N)
136Ø NEXT N
137Ø LET P=17
138Ø LET N=12
139Ø GOSUB 7ØØØ
1399 REM *USE INVERSE PRINT IN *
        *NEXT 2 LINES, EXCEPT *
        *FOR LETTERS L AND M  *
14ØØ PRINT AT 2Ø,Ø;"hit L to con
vert another length"
141Ø PRINT AT 21,7;" hit M for m
enu "
142Ø IF INKEY$="" THEN GOTO 142Ø
143Ø LET W$=INKEY$
144Ø IF W$="L" THEN GOTO 131Ø
145Ø IF W$="M" THEN GOTO 1Ø
146Ø GOTO 142Ø
2ØØØ REM ***area***
2Ø1Ø LET SQC=2.471Ø533
2Ø5Ø DIM A$(15,13)
2Ø6Ø DIM B(15)
2Ø7Ø LET A$(1)="SQUARE MM"
2Ø8Ø LET B(1)=SQC/1E9
2Ø9Ø LET A$(2)="SQUARE CM"
21ØØ LET B(2)=SQC/1E7
211Ø LET A$(3)="SQUARE METRES"
212Ø LET B(3)=SQC/1ØØØ
213Ø LET A$(4)="SQUARE KM"
214Ø LET B(4)=SQC*1ØØØ
215Ø LET A$(5)="CENTIARES"
216Ø LET B(5)=SQC/1ØØØ
217Ø LET A$(6)="ARES"
218Ø LET B(6)=SQC/1Ø
219Ø LET A$(7)="HECTARES"
22ØØ LET B(7)=SQC*1Ø
221Ø LET A$(8)="SQUARE INCHES"
222Ø LET B(8)=1/627264
223Ø LET A$(9)="SQUARE FEET"
224Ø LET B(9)=1/4356
225Ø LET A$(1Ø)="SQUARE YARDS"
226Ø LET B(1Ø)=1/484
227Ø LET A$(11)="PERCHES"
228Ø LET B(11)=1/16
229Ø LET A$(12)="SQUARE CHAINS"
23ØØ LET B(12)=1
231Ø LET A$(13)="ROODS"
232Ø LET B(13)=2.5
233Ø LET A$(14)="ACRES"
234Ø LET B(14)=1Ø
235Ø LET A$(15)="SQUARE MILES"
236Ø LET B(15)=64ØØ
237Ø CLS
238Ø PRINT TAB 11;"A R E A"
239Ø PRINT TAB 1Ø;L$( TO 9)
24ØØ FOR N=1 TO 15
241Ø PRINT AT N+1,Ø+(1 AND N<=9)
;N;".     ";A$(N)
242Ø NEXT N
```

```
2430 LET P=17
2440 LET N=15
2450 GOSUB 7000
2459 REM *USE INVERSE PRINT IN *
         *NEXT 2 LINES, EXCEPT *
         *FOR LETTERS A AND M  *
2460 PRINT AT 20,0;" hit A to co
nvert another area "
2470 PRINT ;"          hit M for m
enu            "
2480 IF INKEY$="" THEN GOTO 2480
2490 LET W$=INKEY$
2500 IF W$="A" THEN GOTO 2370
2510 IF W$="M" THEN GOTO 10
2520 GOTO 2480
3000 REM ***volume***
3010 LET CI=6.1025472
3030 DIM A$(7,21)
3040 DIM B(7)
3050 LET A$(1)="CUBIC MILLIMETRE
S"
3060 LET B(1)=CI/1E5
3070 LET A$(2)="CUBIC CENTIMETRE
S"
3080 LET B(2)=CI/100
3090 LET A$(3)="CUBIC DECIMETRES
"
3100 LET B(3)=CI*10
3110 LET A$(4)="CUBIC METRES (ST
ERES)"
3120 LET B(4)=CI*1E4
3130 LET A$(5)="CUBIC INCHES"
3140 LET B(5)=1
3150 LET A$(6)="CUBIC FEET"
3160 LET B(6)=1728
3170 LET A$(7)="CUBIC YARDS"
3180 LET B(7)=46656
3190 CLS
3200 PRINT TAB 9;"V O L U M E"
3210 PRINT TAB 8;L$( TO 13)
3220 FOR N=1 TO 7
3230 PRINT AT N+3,0;N;".     ";A$
(N)
3240 NEXT N
3250 LET P=13
3260 LET N=7
3270 GOSUB 7000
3279 REM *USE INVERSE PRINT IN *
         *NEXT 2 LINES, EXCEPT *
         *FOR LETTERS V AND M  *
3280 PRINT AT 20,0;"hit V to con
vert another volume"
3290 PRINT TAB 7;" hit M for men
u "
3300 IF INKEY$="" THEN GOTO 3300
3310 LET W$=INKEY$
3320 IF W$="V" THEN GOTO 3190
3330 IF W$="M" THEN GOTO 10
3340 GOTO 3300
```

```
4000 REM ***liquid capacity***
4010 LET PT=1.7598049
4020 DIM A$(8,12)
4030 DIM B(8)
4040 LET A$(1)="MILLILITRES"
4050 LET B(1)=PT/1000
4060 LET A$(2)="CENTILITRES"
4070 LET B(2)=PT/100
4080 LET A$(3)="LITRES"
4090 LET B(3)=PT
4100 LET A$(4)="FLUID OUNCES"
4110 LET B(4)=1/20
4120 LET A$(5)="GILLS"
4130 LET B(5)=1/4
4140 LET A$(6)="PINTS"
4150 LET B(6)=1
4160 LET A$(7)="QUARTS"
4170 LET B(7)=2
4180 LET A$(8)="GALLONS"
4190 LET B(8)=8
4200 CLS
4210 PRINT " L I Q U I D  C A P
 A C I T Y"
4220 PRINT L$
4230 FOR N=1 TO 8
4240 PRINT AT N+2,0;N;".     ";A$
(N)
4250 NEXT N
4260 LET P=13
4270 LET N=8
4280 GOSUB 7000
4289 REM *USE INVERSE PRINT IN *
         *NEXT 3 LINES, EXCEPT *
         *FOR LETTERS L AND M  *
4290 PRINT AT 19,3;" hit L to co
nvert another "
4300 PRINT TAB 7;" liquid capaci
ty "
4310 PRINT TAB 7;" hit M for  me
nu "
4320 IF INKEY$="" THEN GOTO 4320
4330 LET W$=INKEY$
4340 IF W$="L" THEN GOTO 4200
4350 IF W$="M" THEN GOTO 10
4360 GOTO 4320
5000 REM ***weight***
5010 LET P=2.2046224
5020 DIM A$(11,13)
5030 DIM B(11)
5040 LET A$(1)="MILLIGRAMS"
5050 LET B(1)=P/1E6
5060 LET A$(2)="GRAMS"
5070 LET B(2)=P/1000
5080 LET A$(3)="KILOGRAMS"
5090 LET B(3)=P
5100 LET A$(4)="TONNES"
5110 LET B(4)=P*1000
5120 LET A$(5)="DRAMS"
5130 LET B(5)=1/256
```

```
5140 LET A$(6)="OUNCES"
5150 LET B(6)=1/16
5160 LET A$(7)="POUNDS"
5170 LET B(7)=1
5180 LET A$(8)="STONES"
5190 LET B(8)=14
5200 LET A$(9)="QUARTERS"
5210 LET B(9)=28
5220 LET A$(10)="HUNDREDWEIGHT"
5230 LET B(10)=112
5240 LET A$(11)="TONS"
5250 LET B(11)=2240
5260 CLS
5270 PRINT TAB 9;"W E I G H T"
5280 PRINT TAB 8;L$( TO 13)
5290 FOR N=1 TO 11
5300 PRINT AT N+3,0+(1 AND N<=9)
;N;".    ";A$(N)
5310 NEXT N
5320 LET P=16
5330 LET N=11
5340 GOSUB 7000
5349 REM *USE INVERSE PRINT IN *
        *NEXT 2 LINES, EXCEPT *
        *FOR LETTERS W AND M  *
5350 PRINT AT 20,0;"hit W to con
vert another weight"
5360 PRINT TAB 7;" hit M for men
u "
5370 IF INKEY$="" THEN GOTO 5370
5380 LET W$=INKEY$
5390 IF W$="W" THEN GOTO 5260
5400 IF W$="M" THEN GOTO 10
5410 GOTO 5370
6000 REM ***temperature***
6010 DIM A$(3,28)
6020 LET A$(1)="DEGREES CENTIGRA
DE (CELSIUS)"
6030 LET A$(2)="DEGREES FAHRENHE
IT"
6040 LET A$(3)="KELVIN"
6045 CLS
6050 PRINT TAB 5;"T E M P E R A
T U R E"
6060 PRINT TAB 4;L$( TO 23)
6070 FOR N=1 TO 3
6080 PRINT AT N+4,0;N;". ";A$(N)
6090 NEXT N
6100 PRINT AT 11,0;"FROM ?"
6110 INPUT F
6120 IF F<1 OR F>3 OR F<>INT F T
HEN GOTO 6110
6130 IF F=1 THEN PRINT AT 11,5;"
 ",,A$(1)
6140 IF F<>1 THEN PRINT AT 11,5;
A$(F)
6150 PRINT AT 13,0;"TO ?"
6160 INPUT T
6170 IF T<1 OR T>3 OR T<>INT T T
HEN GOTO 6160
6180 PRINT AT 13,3;A$(T)
6190 PRINT AT 15,0;"INPUT TEMPER
ATURE IN"
6200 PRINT A$(F)
6210 INPUT X
6220 PRINT AT 11,0;S$;S$;S$;S$;S
$;S$
6230 PRINT AT 11,0;X;AT 12,0;A$(
F)
6240 PRINT AT 14,4;"IS EQUAL TO"
6250 IF F=2 THEN LET X=(X-32)*(5
/9)
6260 IF F=3 THEN LET X=X-273.15
6270 IF T=2 THEN LET X=X*9/5+32
6280 IF T=3 THEN LET X=X+273.15
6290 PRINT AT 16,0;INT (X*100+0.
5)/100
6300 PRINT A$(T)
6309 REM *USE INVERSE PRINT IN *
        *NEXT 3 LINES, EXCEPT *
        *FOR LETTERS T AND M  *
6310 PRINT AT 19,3;" hit T to co
nvert another "
6320 PRINT TAB 9;" temperature "
6330 PRINT TAB 7;" hit M for  me
nu "
6340 IF INKEY$="" THEN GOTO 6340
6350 LET W$=INKEY$
6360 IF W$="T" THEN GOTO 6045
6370 IF W$="M" THEN GOTO 10
6380 GOTO 6340
7000 PRINT AT P,0;"FROM ?"
7010 INPUT F
7020 IF F<1 OR F>N OR F<>INT F T
HEN GOTO 7010
7030 PRINT AT P,5;A$(F)
7040 PRINT "TO ?"
7050 INPUT T
7060 IF T<1 OR T>N OR T<>INT T T
HEN GOTO 7050
7070 PRINT AT P+1,3;A$(T)
7080 PRINT "INPUT NO. OF ";A$(F)
7090 INPUT X
7100 IF X=0 THEN GOTO 7090
7110 PRINT AT P,0;S$;S$;S$;S$
7120 PRINT AT P,3;X;" ";A$(F)
7130 PRINT TAB 5;"IS EQUAL TO"
7140 LET Y=X*(B(F)/B(T))
7150 LET Z=INT (LN Y/LN 10)
7160 LET Y=INT (Y*10**(6-Z)+0.5)
/10**(6-Z)
7170 LET Y$=STR$ Y
7180 IF Y$(1)="." THEN LET Y$="0
"+Y$
7190 PRINT TAB 3;Y$;" ";A$(T)
7200 RETURN
8000 SAVE "CONVERSION"
8010 RUN
```

Club corner



Looking for a Club in your area? You might just find it here . . .

The ZX Postal Club

Dear ZX Computing,
Would you be so kind as to publish this letter. I would like to start a correspondence club with English-speaking Spectrum users in other European countries who do not have ready access to a club in their area, or who would just like to correspond with someone in Luxembourg.

People who would like to have more information about this club should write to the address given below.
Yours faithfully

Neil Henderson,
6, Rue Belle-vue,
Strassen.
Tel: 318379

Western Australia ZX Group

Dear ZX Computing,
This is from the WA ZX Group. We are about 50 devotees of ZX's and Spectrums. We meet monthly on the third Monday at the Melville High School, between Perth and Fremantle.

Anyone interested is invited to contact the Honorary Secretary, Garth Gregson, on 335 1671. Thank you for ZX Computing, it is read avidly by all our members.

Garth Gregson,
34, Chester Street,
South Fremantly,
Western Australia 6162.

The Sultan Computer Club.

Dear ZX Computing,
A plug for our computer club at work. Anybody in other naval establishments who would like to contact us, please write to me, CPO Dibsdale (Chairperson) or CPO Perkins (secretary) at the address below.
Yours faithfully

CPO Dibsdale,
Rutherford Block,
HMS Sultan,
Gosport,
Hants.

ZX 80 Club Hameln

Dear Ladies and Gentlemen!
Please, allow me to introduce our club to you. The Club was founded by Mr Thomas Jenczyk under the name "ZX-80 Club Hameln". Our 160 members come from the Netherlands, Germany, Austria and Switzerland. Although our club is concerned with the Sinclair computers ZX 80, ZX 81 and Spectrum, many members work with other computers — the range is from little single-boarders to multi-user-systems.

Mr Jenczyk now has to spend most of his time on other things so I, Aribert Deckers, have taken over the coordination of the club. Please, take our new address to your files.
Yours sincerely

Aribert Deckers,
Postfach 967,
D-7000 Stittgart 1,
West Germany.

Someone, Somewhere, in Somerset?

Dear ZX Computing,
I wish to form a Sinclair computer club in the Taunton area of Somerset. We would meet once a month to exchange ideas and news, and a magazine would be published every three months. The club will probably be called 'ZX Bug Breaker' and a membership fee of £2.00 per year to cover the cost of the magazine will be harged. The owner of any Sinclair computer, including the QL and the ZX80 would be admitted and memvers would not be obliged to exchange any idea they had or to attend meetings. I would like a number of about 30 people in the club and Spectrum owners are especially welcome. Please would you publish this in your club corner.

Thank you for helping me in this matter. I think your magazine is the best on the market and would recommend it to anyone. For more details of the club please contact me at the address below or by phone at: Taunton (0823) 86291. All the best with the magazine,
Yours sincerely

Alexander Poole,
51 Whitmore Road,
Taunton, TA2 6EA
Somerset.

Chinnor and Thame

Dear ZX Computing,
In February 1984, a group of local computer enthusiasts met with a view to establishing a computer user group in the area. As one of the enthusiasts was Alan Head, Headmaster of Mill Lane School, it was soon evident that we would have the support of Mill Lane Association in terms of providing us with facilities. It therefore seemed appropriate the group should operate under the title Mill Lane Association Computer Group, and our inaugural meeting was held on 19 March 1984 in Mill Lane School Hall.

Since the inception of the Group, we have been determined to operate in accordance with two fundamental principles. Firstly, we welcome computer enthusiasts of all ages and abilities, from the parent who wants to find out more about computers and computing to the lads (or lassies!) who want to show others what they have learnt from using their machines. Secondly, we welcome machines of all types and will not restrict our activities to any particular make or model.

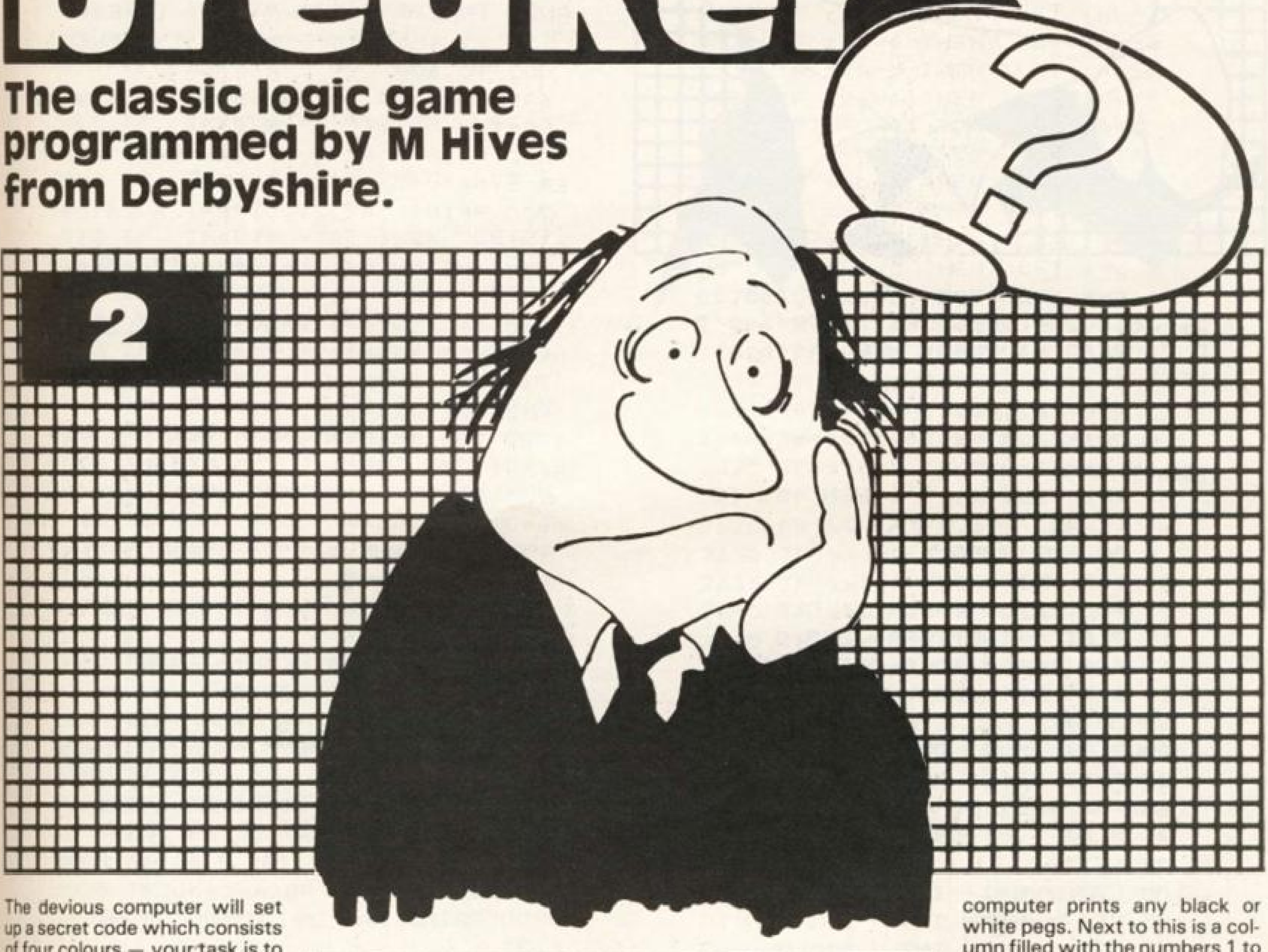
We intend to organise the Group's activities to cater as much as possible for the needs of the majority of our members — so far we have arranged a discussion on computer applications led by the well known computer software and book author Trevor Toms, a "Machines Evening" where a number of different machines have been brought along so that members can see the strengths and weaknesses of each one, and a "Competition Evening" where programming competitions are set with prizes for the winners. We hope to plan more talks, discussion groups, games evenings and possibly organised visits to computer exhibitions and manufacturers in the future. It is also our intention to establish a group library containing books and software for group members to borrow — we already have a "Free Box" containing unwanted magazines donated by members.

Subscriptions for the remainder of 1984 are £7.00 for family membership, £5.00 for individual membership and £3.00 for those under 16 years. The Group will meet on Thursday 3 May, Thursday 17 may, Friday 1 June, Thursday 14 June and fortnightly thereafter on Thursdays at 7.30 pm in Mill Lane School Hall.

For further information, please contact Bryan McAlley on Kingston Blount 52426, Mike Oswell on Kingston Blount 54002 or Len Tack on Thame 2367.

Cipher breaker

The classic logic game programmed by M Hives from Derbyshire.



The devious computer will set up a secret code which consists of four colours — your task is to decipher the code and beat the machine.

After you have entered all four colour codes one at a time, the computer will give you an indication of how well you have done. For a correct colour in the correct position the computer will award you a black peg, and for a correct colour but in the wrong place you are given a white peg.

Unfortunately you do not know which peg relates to which colour, so you will have to try various combinations using your superior human logic to find the combination.

On the far left of the screen is displayed a column in which the computer prints any black or white pegs. Next to this is a column filled with the numbers 1 to 10: this is the number of tries you have had and the number left!

The remainder of the screen is taken up with the four columns in which your colour codes are displayed.

This is a good game which features good use of sound and colour and a large, clear screen display. We enjoyed playing this game in the office but wondered why we hadn't yet seen a version of this game which lets the computer attempt to decode a human secret code.

If anyone out there has a program which uses a logical system of computer decoding then how about letting us see it?

Program lines

1-6	Sets up UDGs and gives instructions
11-25	Sets up the board giving ten attempts (also lines 199 to 203)
100-115	Sets up the code in numerical form
500-950	Asks the player for colours, changes it to a number and checks against the code.
1005-1035	Checks for any white pegs
1455-1470	Checks for any black pegs
1480	Checks for a correct answer
1500	Gives you another attempt using a FOR/NEXT loop
2000-2020	Part of the winning routine
2500-2660	Turns the computers code from numbers into colours and prints the correct answer at the top of the screen



```
   1 CLS : DATA 24,60,126,255,25
5,126,60,24: RESTORE : FOR a=0 T
O 7: READ b: POKE  USR "A"+a,b:
NEXT a
   3 PLOT 62,78: DRAW 90,0
   4 PRINT  AT 21,0;"Press any k
ey to continue"
   5 IF  INKEY$ ="" THEN  GO TO
5
   6 GO SUB 9900
  10 LET x=1
  11 BORDER 5: PAPER 5: CLS
  15 PLOT 0,168: DRAW 255,0
  17 FOR f=152 TO 8 STEP -16
  18 PLOT 0,f: DRAW 255,0
  19 NEXT f
  20 PLOT 35,0: DRAW 0,168
  22 PLOT 57,0: DRAW 0,168
  23 FOR f=57 TO 255 STEP 40
  24 PLOT f,8: DRAW 0,160
  25 NEXT f
 100 LET zxa=1+ INT ( RND *6)
 105 LET zxb=1+ INT ( RND *6)
 110 LET zxc=1+ INT ( RND *6)
 115 LET zxd=1+ INT ( RND *6)
 199 LET b=1
 200 FOR a=1 TO 20 STEP 2
 201 PRINT  AT a,5; INK 2;b
 202 LET b=b+1
 203 NEXT a
 500 FOR b=1 TO 10
 600 PRINT  AT 21,0;"Which colcu
r ?"
 605 IF  INKEY$ ="" THEN  GO TO
605
 610 IF  INKEY$ ="b" THEN  LET i
nk=1: LET ua=1
 615 IF  INKEY$ ="r" THEN  LET i
nk=2: LET ua=2
 620 IF  INKEY$ ="w" THEN  LET i
nk=7: LET ua=3
 625 IF  INKEY$ ="m" THEN  LET i
nk=3: LET ua=4
 630 IF  INKEY$ ="y" THEN  LET i
nk=6: LET ua=5
 635 IF  INKEY$ ="g" THEN  LET i
nk=4: LET ua=6
 640 IF  INKEY$  <> "b" AND  INK
EY$  <> "m" AND  INKEY$  <> "r"
AND  INKEY$  <> "w" AND  INKEY$
 <> "y" AND  INKEY$  <> "g" THEN
  GO TO 600
 645 FOR s=0 TO 60: BEEP .005,s:
 NEXT s
 650 PRINT  AT x,9; INK ink; PAP
ER 5;"A"
 700 PRINT  AT 21,0;"Which colou
r ?"
 705 IF  INKEY$ ="" THEN  GO TO
705
 710 IF  INKEY$ ="b" THEN  LET i
nk=1: LET ub=1
 715 IF  INKEY$ ="r" THEN  LET i
nk=2: LET ub=2
 720 IF  INKEY$ ="w" THEN  LET i
nk=7: LET ub=3
 725 IF  INKEY$ ="m" THEN  LET i
nk=3: LET ub=4
 730 IF  INKEY$ ="y" THEN  LET i
nk=6: LET ub=5
 735 IF  INKEY$ ="g" THEN  LET i
nk=4: LET ub=6
 740 IF  INKEY$  <> "b" AND  INK
EY$  <> "m" AND  INKEY$  <> "y"
AND  INKEY$  <> "g" AND  INKEY$
 <> "r" AND  INKEY$  <> "w" THEN
  GO TO 700
 745 FOR s=0 TO 60: BEEP .005,s:
 NEXT s
 750 PRINT  AT x,14; INK ink; PA
PER 5;"A"
 800 PRINT  AT 21,0;"Which colou
r ?"
 805 IF  INKEY$ ="" THEN  GO TO
805
 810 IF  INKEY$ ="b" THEN  LET i
nk=1: LET uc=1
 815 IF  INKEY$ ="r" THEN  LET i
nk=2: LET uc=2
 820 IF  INKEY$ ="w" THEN  LET i
nk=7: LET uc=3
 825 IF  INKEY$ ="m" THEN  LET i
nk=3: LET uc=4
 830 IF  INKEY$ ="y" THEN  LET i
nk=6: LET uc=5
 835 IF  INKEY$ ="g" THEN  LET i
nk=4: LET uc=6
 840 IF  INKEY$  <> "b" AND  INK
```

```
EY$  <> "r" AND  INKEY$  <> "w"
AND  INKEY$  <> "m" AND  INKEY$
 <> "y" AND  INKEY$  <> "g" THEN
  GO TO 800
 845 FOR s=0 TO 60: BEEP .005,s:
 NEXT s
 850 PRINT  AT x,19; INK ink; PA
PER 5;"A"
 900 PRINT  AT 21,0;"Which colou
r ?"
 905 IF  INKEY$ ="" THEN  GO TO
905
 910 IF  INKEY$ ="b" THEN  LET i
nk=1: LET ud=1
 915 IF  INKEY$ ="r" THEN  LET i
nk=2: LET ud=2
 920 IF  INKEY$ ="w" THEN  LET i
nk=7: LET ud=3
 925 IF  INKEY$ ="m" THEN  LET i
nk=3: LET ud=4
 930 IF  INKEY$ ="y" THEN  LET i
nk=6: LET ud=5
 935 IF  INKEY$ ="g" THEN  LET i
nk=4: LET ud=6
 940 IF  INKEY$  <> "b" AND  INK
EY$  <> "r" AND  INKEY$  <> "w"
AND  INKEY$  <> "m" AND  INKEY$
 <> "y" AND  INKEY$  <> "g" THEN
  GO TO 900
 945 FOR s=0 TO 60: BEEP .005,s:
 NEXT s
 950 PRINT  AT x,24; INK ink; PA
PER 5;"A"
 960 PRINT  AT 21,0;"
                   "
1005 IF ua=zxb OR ua=zxc OR ua=z
xd THEN  PRINT  AT x,2; PAPER 5;
 INK 7;"A"
1015 IF ub=zxa OR ub=zxc OR ub=z
xd THEN  PRINT  AT x,3; INK 7; P
APER 5;"A"
1025 IF uc=zxa OR uc=zxb OR uc=z
xd THEN  PRINT  AT x+1,2; INK 7;
"A"
1035 IF ud=zxa OR ud=zxb OR ud=z
xc THEN  PRINT  AT x+1,3; INK 7;
"A"
1455 IF ua=zxa THEN  PRINT  AT x
,2; PAPER 5; INK 0;"A"
1460 IF ub=zxb THEN  PRINT  AT x
,3; INK 0; PAPER 5;"A"
1465 IF uc=zxc THEN  PRINT  AT x
+1,2; INK 0;"A"
1470 IF ud=zxd THEN  PRINT  AT x
+1,3; INK 0;"A"
1475 LET x=x+2
1480 IF ua=zxa AND ub=zxb AND uc
=zxc AND ud=zxd THEN  GO TO 2000
1500 NEXT b
1510 GO TO 2500
2000 FOR s=0 TO 7 STEP .3
2010 PRINT  AT 0,20; INK s;"CORR
ECT"
2015 BEEP .004,s*8
2020 NEXT s
2500 PRINT  AT 0,0;"Answer=
                   "
2510 IF zxa=1 THEN  LET ink=1
2515 IF zxa=2 THEN  LET ink=2
2520 IF zxa=3 THEN  LET ink=7
2525 IF zxa=4 THEN  LET ink=3
2530 IF zxa=5 THEN  LET ink=6
2535 IF zxa=6 THEN  LET ink=4
2540 PRINT  AT 0,9; INK ink;"A"
2550 IF zxb=1 THEN  LET i=1
2555 IF zxb=2 THEN  LET i=2
2560 IF zxb=3 THEN  LET i=7
2565 IF zxb=4 THEN  LET i=3
2570 IF zxb=5 THEN  LET i=6
2575 IF zxb=6 THEN  LET i=4
2580 PRINT  AT 0,14; INK i;"A"
2590 IF zxc=1 THEN  LET i=1
2595 IF zxc=2 THEN  LET i=2
2600 IF zxc=3 THEN  LET i=7
2605 IF zxc=4 THEN  LET i=3
2610 IF zxc=5 THEN  LET i=6
2615 IF zxc=6 THEN  LET i=4
2620 PRINT  AT 0,19; INK i;"A"
2630 IF zxd=1 THEN  LET i=1
2635 IF zxd=2 THEN  LET i=2
2640 IF zxd=3 THEN  LET i=7
2645 IF zxd=4 THEN  LET i=3
2650 IF zxd=5 THEN  LET i=6
2655 IF zxd=6 THEN  LET i=4
2660 PRINT  AT 0,24; INK i;"A"
2670 PRINT  AT 21,0;"Press any k
ey to play again"
2675 IF  INKEY$ ="" THEN  GO TO
2675
2680 CLS : GO TO 10
9900 PAPER 5: CLS : PRINT  AT 2,
0;"'b' -"; INK 1;"Blue"
9901 PRINT '"'r' -"; INK 2;"Red"
9902 PRINT '"'w' -"; INK 7;"Whit
e"
9903 PRINT '"'m' -"; INK 3;"Mage
nta"
9904 PRINT '"'y' -"; INK 6;"Yell
ow"
9905 PRINT '"'g' -"; INK 4;"Gree
n"
9907 INK 0
9910 PRINT '''"When asked ""Whic
h colour ?"" use the above lette
rs"''"Press any key to start"
9915 IF  INKEY$ ="" THEN  GO TO
9915
9920 RETURN
```

BUFFER MICRO SHOP

310 STREATHAM HIGH ROAD, LONDON SW16 6HG
Tel: 01-769 2887

BOOKS, GAMES, SMALL BUSINESS SYSTEMS, EDUCATIONAL, HOUSEHOLD. COLOUR MONITORS, CONNECTING WIRES, COMPUTING AIDS.

THE ORIGINAL "SINCLAIR" SPECIALISTS

FOR 48K SPECTRUM
£5.50
ATHLETE
RUNNING, HURDLES, SHOT....
BEAUTIFUL GRAPHICS,
COMPULSIVE ACTION

EXCLUSIVE SCOOP

£5.95
BRIDGE — NEW VERSION
ALL MACHINE CODE
OUTPLAYS ALL OTHERS

£5.95
POOLS PREDICTION
IN DEPTH ANALYSIS OF BRITISH AND AUSTRALIAN LEAGUES

£5.95
MUSIC
SYNTHESISER *PROG FOR USE*
WITH MOST ADD-ON SOUND BOXES

HARDWARE AND ACCESSORIES, LIGHT PENS, TRACERS, SOUND BOXES, PRINTERS, V.D.U's, INTERFACES, KEYBOARDS, MEMORY EXPANSIONS

VISA – ACCESS – AMERICAN EXPRESS – DINERS CLUB
Open Tues.-Sat. 10.30 a.m. to 5.30 p.m. (closed Monday)

HEARING IS BELIEVING

Tele Sound

SPECTRUM BEEP BOOSTER

TELESOUND is ready built and tested so that you can get the full sound of your Spectrum through any unmodified television set, black and white or colour. The unit is easy to fit in minutes without any previous experience as full fitting instructions are provided.

TELESOUND has been designed using the very latest ultra miniature components so it's size is an **incredible 2 x 1.3 x 1cm. and fits neatly inside** the Spectrum. Three easy push-on connections are made to the Spectrum. The connections are clearly shown in a diagram that comes with full fitting instructions.

TELESOUND FEATURES

- SOUND AND VISION TOGETHER FOR ADDED REALISM.
- BEEP VOLUME CONTROLLABLE FROM A WHISPER TO A ROAR.
- KEYBOARD CLICK CLEARLY HEARD TO ASSIST PROGRAM ENTRY.
- PROGRAMS CAN BE HEARD WHEN BEING LOADED • NO SOLDERING OR CASE CUTTING REQUIRED.
- ADDITIONAL AMPLIFIER NOT NECESSARY • SEPARATE POWER SUPPLY NOT REQUIRED.
- USES NO MEMORY SO WORKS WITH ALL PROGRAMS • COMPATIBLE WITH ALL OTHER ADD-ONS INCLUDING MICRODRIVE

TELESOUND comes complete with easy to follow fitting instructions and costs £9.95 inclusive of post, packing etc. Please state your computer when ordering.

COMPUSOUND

BARCLAYCARD VISA — C T A — Access

DEPT: ZX **32 - 33 LANGLEY CLOSE, REDDITCH, WORCESTERSHIRE. B98 0ET. TELEPHONE: (0527) 21429 (21439 Answerphone)**

Wizard Software

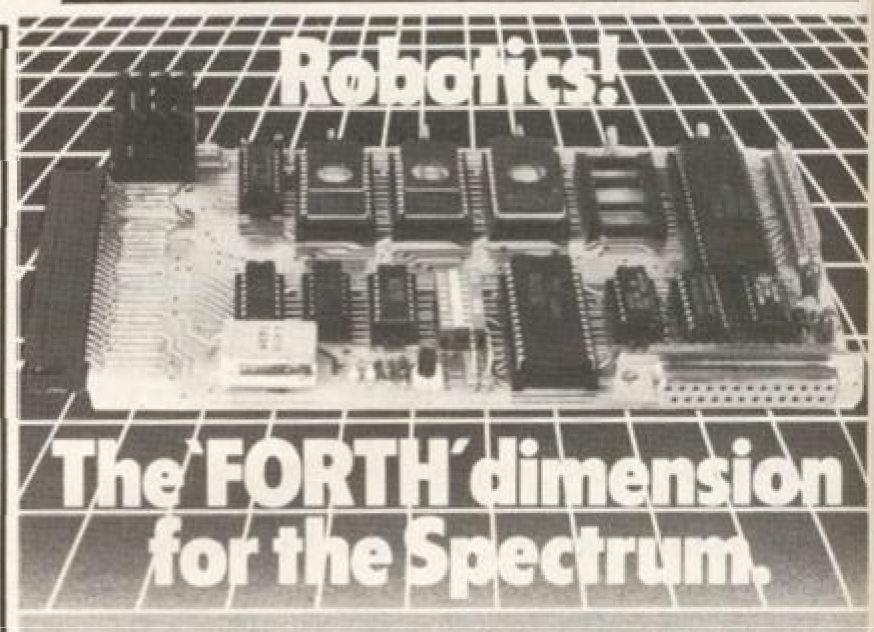


ZX SPECTRUM 48K

BUGS **£5.95**
Superb use of colour and animated graphics make this multiple page machine code Arcade game a joy to play. In this slick and addictive game you have to manipulate BUGS, a software bug, whose task is to lay a track in order to mend faulty components on the printed circuit boards inside your computer. Three board layouts are used in 8 different colour combinations to produce 24 board variations. As well as mending these boards you will have to traverse the cables joining the boards. Just to make things more difficult you are hampered in your task by 3 varieties of demon hardware bugs, GLITCH, SPIKE and HOTSPOT, who chase you around the boards or rush along cables at you. Beware if they catch you, as you dodge around the boards or as you leap over them on the cable, or you will be fried.

SPECTRUM CRICKET **£5.45**
Play either test match or limited over cricket. Each game is a graphical representation, the visual display comprising the cricket oval and participants together with a continually updated score board. Select your teams from the 6 international sides available or choose your own players. The graphics move in the characteristic manner of real players, the batsmen take guard, strike the ball and on deciding to run move between the wickets, the bowlers are seen to deliver the ball, fielders move to field the ball and throw at the wickets or return the ball to the bowler, umpires make the appropriate signals, wickets are seen to fall and the ball is thrown in the air by ecstatic fielders when a batsman is caught. 72 field positions available. 3 types of bowler delivering a variety of balls. At the end of a match you have the option of a hard copy of the score cards on your printer. Game save facility.

Also available: **Time Port 1 £6.95, Evictor £5.95, Fruita £5.45**
All prices inclusive, mail order, cheque or postal orders to:
Wizard Software, Dept. ZX, PO Box 23, Dunfermiline, Fife, KY11 5RW.
Also available from software retailers world wide. Send large S.A.E. (7in. x 5in.) for full program catalogue. Royalties paid for machine code SPECTRUM software.



Now your 16k and 48k Spectrum get 12k of fig-FORTH from a new cartridge that's perfect for robotics.

This piece of genius is the creation of David Husband and it's the only ROM cartridge of its kind available.

It has RS232 and Parallel ports that not only facilitate remote control but are usable from FORTH or BASIC with the parallel also allowing a Centronics printer to be driven.

Due to an Interrupt driven 'Break' key the machine cannot be 'hung-up' and a number of routines and FORTH words are Vectored allowing reconfiguration.

Later in the year a software upgrade will be available which will permit multi-tasking.

Order the Spectrum FORTH I/O Cartridge £59+VAT using the coupon adding £5.75 p&p & insurance (£10 for Europe, £15 outside) or if you want more detailed information, tick that box instead. SUBJECT TO AVAILABILITY

Please send me Spectrum Forth I/O Cartridge. £59+VAT
Tick whether for 16k ☐ or 48k Spectrum ☐
Multi-Forth 81 ☐ ZX81 Forth ROM ☐

Name________
Address________
________Post code________

Send to Skywave Software, 73 Curzon Road, Bournemouth, BH1 4PW, Dorset, England. Tel: (0202) 302385

Skywave SOFTWARE

SPECTRUM FORTH I/O CARTRIDGE

Readers' Reviews

Reviews of software and hardware from the people who use them most — yourselves



Doomsday Castle Fantasy Software Mark Stoneman

To be honest it was the unusual box that attracted me to Fantasy Software's Doomsday Castle more than anything else (it looks more like a book than the customary cassette box that I am used to). One advantage of this feature is that it allows the use of a removable instruction sheet which is easier to glance at than an inlay card. The literature enclosed, although long, is both thorough and witty and worth reading. However, I am glad that the oddity of the wrapping of this game persuaded me to part with my £6.50 as it is one of the best games I've seen in a long time.

The villain of this piece is the infinitely evil Scarthax who has found six ancient magic stones which enable him to control the universe. He has imprisoned them in Doomsday Castle but has placed them in separate rooms because if all six are brought together there will be a chain reaction causing instant anihilation.

The program loads in 3 minutes 35 seconds and confronts you with a menu of controls, there are seven to choose from: the Fuller, Kempston, Sinclair and cursor joysticks or a choice of three keyboard layouts. Although the latter is a good idea, user-definable keys would have been better.

You control the hero, Ziggy, who flies about in an exploratory Capsule armed only with a Mini Missile Launcher, with which he can destroy Urks, Neucloids, Orphacs and doors. However after sixteen door-shots your MML will run out of Theta Radiation which gives sufficient power to destroy and in order to recharge you must kill eight Urks, absorbing their radiation. At the bottom of the screen is shown your shield power. This commences at 1000 and gradally decreases as you collide with and are shot by the various inhabitants of the castle (when this reaches zero Ziggy will shortly die).

Doomsday Castle consists of twenty five halls and 49 connecting passages. Each hall contains four doors, two lifts (which you cannot use) and an antechamber. One of the doors you will already have come through and so will not be able to use it again (when this occurs the door is coloured blue). The only way to get through the exit doors which are coloured red, is by blasting them away; this is quite an involved process! The Antechamber stands at the bottom and contains either a crystal or an Ancient Stone. A crystal will increase your Shield Power whilst the stones will add to your score. On top of the Antechamber sits a Googly Bird, one of which inhabits every hall. Its main aim in life is to sleep but if woken it will fly up to the ceiling and unleash a thread which will greatly sap your shield power. The only way the above bird can be woken is if an Urk falls on its head: you must therefore prevent them doing this.

As you explore the castle you will find that Urks are the commonest form of life there; they take over fifty different forms (one type in each room). Urks will atomize should they come into contact with absolutely anything. However, they will drain your shield power.

A Garthog sits on each lift firing indiscriminately across the hall. They constantly grin except when shooting them, when they bear an amusing frown.

Should you manage to shoot your way out of a hall you will be sucked along a narrow corridor where the name of the hall you have just left, your score, the high score, the stones collected and your shield power are displayed. You are then plummetted into a larger passage where you are locked in with Neucloids, Orphacs and, of course, Urks. Neucloids are very primitive, single-cell creatures which multiply at an alarming rate and are very difficult to destroy. Orphacs are programmed robots which will bounce off the walls until they hit you. There are two doors in the passage, one of which you will just have come through and the other through which you must exit. However, unlike in the halls the exit is operated by a time lock so you must wait until it opens, avoiding the alien wildlife and losing as little shield power as possible.

At the end of each game your score is shown together with a code and if you wish you can send both of these into Fantasy and they may be published in one of the two high score tables published annually. One small point, in this game points are acquired by the number of Ancient Stones gained and not by the number of aliens killed: interesting.

Doomsday Castle is available for the 48K Spectrum from Fantasy Software. Its superb graphics, sound and absorbing playability make it well worth £6.50. I highly recommend you to buy it.

Scope Computer Graphics Language ISP Tim Pickford-Jones

For those hardy souls wishing to write arcade-ish games to run at phenomenal speed with slick graphics and dynamic sound there is really no alternative yet to joining the ice-cool whizz kids whose daily diet of DJNZs and EX (SP), HLs send shivers of incomprehension down the computer corridors. However, with the SCOPE a small step towards a human interface with the machine has been made.

What you have to do, after perusing the 48 page instruction book-cum-dictionary of SCOPE terms, is to type a lot of single statement REM lines as per BASIC, then type in the magic command PRINT USR 60450 and hey presto (if you've done it all O.K.) the SCOPE compiles wads of unseen code which will certainly run a lot faster than the equivalent BASIC. The SCOPE system uses 31 key words, deliberately unlike their BASIC counterparts, such as "PUT" instead of "PRINT AT" which nearly all have certain arguments like line and column numbers, colour numbers, pitch and duration etc.

If, however, you need to know the value of PI to 16 places or the 1,000th prime then stop here. The SCOPE has a very limited arithmetical vocabulary; only increase and decrease on 1-byte integers and add for the 2-byte integers intended for scores. What you do get is a system for producing chunks of code which could be used as a complete programme or part of a hybrid. Passing values on from BASIC to SCOPE and vice-versa is not mentioned in the book, nor is it at all obvious how it could be done. Suffice it to say that you will have to ▶

search around until you find the SCOPE variable store and do some nifty PEEKing and POKEing from BASIC.

The tape loads easily in 35 seconds and produces a suite of subroutines (one of which is the compiler itself) from address 60000 to the end of the computer. It then NEWS the BASIC area with the ramtop set at 59999. So, naturally you would write the eventual code at some point below this. Having successfully compiled some code the clever programme tells you the next free byte after the chunk it has just done. One slight drawback here is that the whole of the SCOPE programme has to be in the computer as well as the code you have written. This is because the resulting code calls the many subroutines contained in the SCOPE. This is not really all that bad as the routines have to exist somewhere, but there are other compilers that provide copies of the required routines in the compiled code and don't bother with the ones you haven't used, so the finished article is independent of the compiler.

A word to those who have managed to bust the bank (not to mention Uncle's waiting list) and have Microdrives. The SCOPE as it stands will encounter problems compiling your BASIC if it does not begin at 23755, which of course it will not if you have recently used your drives. All is not lost, however, and three quick POKEs into the SCOPE and re-recording of the code before use will solve the problem once and for all. First load the SCOPE then, as direct commands, enter the following: POKE 60459, 42: POKE 60460,83: POKE 60461,92. This now tells the SCOPE to find the beginning of the BASIC from the Spectrum's system variables instead of slavishly imagining that all BASIC must start at 23755. Now SAVE "Scope" CODE 60000,5535 to a new piece of tape and there you have it. A simple and very useful modification. You will need to make a copy of the SCOPE BASIC loader, or write one yourself, remembering to set the ramtop to 59999 before loading "Scope"CODE.

Negative aspects of the SCOPE are that text printing leads to some rather long winded code, the Fine-scroll (by one pixel) works in all directions, but leaves the attributes behind, and coarse scroll only works upwards, but does include the attributes. There is a need to end each BASIC statement with a colon, which to my mind is redundant as Sinclair lines have an end of line marker and a 2-byte length of line indicator. It is a pity that the SCOPE writers did not use this instead of requiring the user to put in an additional marker. Error trapping is quite good, but having trapped mistakes in the BASIC you are not given much of an idea as to where the error is or what is wrong. All you get is "Perameter Error".

ON the plus side, use of the SCOPE can encourage more structure in programming without the user realizing it. It may tend towards less spaghetti in future BASIC. For the more adventurous it can provide a selection of useful code routines which can be copied or amended for specific purposes, and there are some interesting solutions to the problem of indirect and implied addressing in relocatable code! As for that claim made by the makers that, "SCOPE has one of the great advantages of FORTH without the comcomitant disadvantages..." read carefully. They do not mention what these disadvantages are, and the ability to write subroutines which can call other subroutines is not confined to FORTH; the reference here is a little spacious.

Neatly packaged, well documented, SCOPE is a slightly friendly bridge between the real world of invaders and galaxians and the innermost workings of the micro. At £11.95 it's rather expensive, but as the blurb says: "Now; unlock your imagination."

Lunar Jetman
Ultimate
Brian Buckley

Lunar Jetman by ULTIMATE PLAY THE GAME takes 4½ minutes to LOAD, which is longer than most other games for the 48K Spectrum. In common with all other ULTIMATE games, a picture of the cassette insert illustration is built up on the screen during LOADing. Gazing at this helps pass the LOADing time as does reading the scenario and instructions contained on the insert.

Starting with overtones of the 'Six Million Dollar Man' it transpires that Jetman (yes, the same little person who starred in Jetpac) has crash landed on 'a strange undiscovered world' where he discovers the inhabitants are about to attack his beloved Earth, and destroy us poor unsuspecting souls. Bravely, our little hero sets about destroying all of the enemy missile installations he can find. All he has to aid him in this mammoth task is his Hyperglide Moon Rover (Sinclair's electric car?), an unlimited supply of bombs and bridging platforms, and of course, his faithful old Quad Photon Laser Phaser. This may seem a fairly straightforward exercise, but be warned, it's not!



The first problem is that Jetman can't find the operating manuals for anything, so he doesn't know how to use bombs, teleporters, detachable gun or the bridging platforms and indeed doesn't even know where to get the platforms! After playing the game for a while, things will gradually become clearer, but you will probably have lost countless men and suffered severe frustration by this time! The teleporters are the cylindrical objects which sit on the surface of the planet, and these allow Jetman to transport himself (and indeed anything which he happens to be carrying) to another point on the planet. This part brings back fond memories of the classic 'Star Trek' series (beam me up Scottie!).

The other problem is the inevitable aliens, but more of these later. Once the game has LOADed, a menu appears allowing the choice of a one or two player game and the method of control, i.e. keyboard, cursor joystick or Kempston Joystick. Having made your selection, the screen changes to show the landscape at the bottom with the lunar rover sitting close to a bomb (complete with fuse!). At the top of the screen, one and two player scores are displayed, along with lives left, the number of bases destroyed, direction indicators (which show the direction of the missile installation and lunar rover), a guage showing the remaining fuel in your Hydrovac Jetpac, one for the time remaining until the missiles are launched and finally, in the middle, the hi-score, which stands at an unbelievable 105140! Don't despair though, it can be beaten.

Jetman appears close to the lunar rover a few seconds after the main screen appears, so unless you have your fingers at the ready, he'll probably be flattened by something before he's had a chance to move! At the start of the game there are three types of alien flying about, all with their own pattern of behaviour. As the number of destroyed bases increases, so does the number of alien species, and after blowing four bases to pieces, things get decidedly hair-raising to say the least! Up until now, it's been easy! Having obliterated ten bases you will find yourself dealing with no less than ten different types of alien, all of which will cause the instant demise of your Jetman should he touch them. Of these ten types, three actually follow you, and another shoots it's own missiles!

In the event of not reaching a base in time, you are given a warning informing you that the missiles have been launched. The only way to prevent your lunar rover from being blown up is to fly towards the base until you encounter the missile, and then shoot it (the number of hits required are given in the warning) with your phaser, dodging the aliens in the process!

The game ends when your lunar rover is blown up by a missile, or you lose all of your lives. Incidentally, an extra life is awarded after every four bases.

A hall of fame exists in the program and lists the eight highest scores along with the initials of the players who obtained them. The scores range from 10500 to 105140, and when you score enough points to qualify entry to the chart, your problems aren't over! You'll have to figure out how to place your initials in the chart, as there aren't any instructions to help you!

The choice of control keys has been sensibly made, so that even left-handed people shouldn't have any trouble in using the keyboard. Indeed, it seems to be easier to play the game using the keyboard than it does using a Kempston joystick! In all, there are seven keys to master, controlling movement left, right and up, firing, picking up and dropping objects, getting into and out of the lunar rover and teleporter, and finally, one for hovering. The game can be

paused by pressing the '0' (letter) key, and restarted by pressing this key again.

The graphics are nothing short of superlative, and the smoothness of movement is such that you could be forgiven for thinking that you are watching a cartoon! Sound is excellent, as is the use of colour.

All in all this is a very well written and presented game which is also very addictive. Even the character set has been redefined! Extraordinary attention has been paid to detail (the aerial on Jetman's backpack wobbles when he walks!) making this in my opinion the best computer game of all time. Anyone who buys games for their Spectrum should have Lunar Jetman in their collection. At £5.50 it's a bargain! Full marks to ULTIMATE for a superb piece of programming!!



Cassette 5 Orwin Software Greg Woods

Michael Orwin, or as is now called, Orwin Software, have brought out another of their eight games cassette's for the ZX81. Namely, the cassette 5. Each game on the cassette will be reviewed separately below.

PLANETOIDS. This game is an adaptation of the popular arcade game. The object being to shoot the rocks and other creatures floating in space. You can move and fire in eight directions, and you have a panic (hyperspace) button. The game is fast and extremely enjoyable. It could have easily been sold on it's own for a reasonable amount.

BREAKOUT. Everyone should know how to play this game, but Orwin's version has an added twist; the wall scrolls down to you. I found this much more challenging. My only criticism is that it crashed occasionally. Not very useful.

SPACE RESCUE. This game is not as well written as the last two, as it is rather slow. However, your mission is to rescue six astronauts from the moons surface. On your descent, you must avoid the meteors obscuring your path, and then land on a pad to collect an astronaut. Then you must weave past the aliens and dock with the mothership.

DRAUGHTS. No explanation is needed on how to play this game. This version is fast with three levels.

BYTE-MAN. You guessed it, another Pac-man type game. This one has three mazes, and three ghosts chasing you. I thought this game was fast to start with, but when you've finished a maze, it really gets fast. An excellent game.

DODGEMS. Another 'Pub' favourite in which you can change lanes in your car whilst dodging the computer's car opposing you. This is fast and well written, but not very addictive.

BLITZ. As the name suggests, your mission is to fly over a city dropping bombs so that you can land your craft. The action is somewhat jerky however, and the progem has a strange resemblance to a game in Orwins cassette 4.

MERCHANT. At last a change from the invaders. This is a simulation in which you sail from port to port, buying and selling goods in order to make a profit. You have to beware of greedy tax men (shouldn't we all) and storms as you head towards your retirement. Another great game.

CONCLUSION. The cassette 5 is an excellent buy at £6, for eight great games. If you like arcade games in particular and you've already got cassette 4, then get cassette 5. If not then get cassette 4 first.

Cassette 5 is available by mail order from: MICHAEL ORWIN, 26, BROWNLOW RD., WILLESDEN, LONDON NW10 9QL.

3D Grand Prix D.K. Tronics Jamie Pullen

After two weeks an impressive looking tape came through the post. It is SAVEd on both sides in case of any mishaps, although it loaded first time. After a three minute loading time an excellent title page appeared with a racing car on a DK Tronics stand. When you press a key as instructed it zooms off smoothly and the first page of instructions appear. After you have read the **five** pages of instructions (very detailed) you wait thirty seconds for it to generate a new track layout.

When you first see the screen you just have to fall back in your chair and admire the graphics. Occupying the bottom of the screen are your instruments, telling you what gear you are in (a total of six gears), how many revs you are doing (15,000 possible), your position in the race, your speed (maximum speed of 180), fuel left (you start with 50 gallons) and finally what lap you are on.

When you start the game you are in 18th position. On the left of the screen is an overhanging lamp post which tells you when you can go or not. When the light shows red you can't go, and believe it or not, when it shows green you can go. When the other cars have sped away you see the chequered road. Depending how fast you are going it soon disappears and leaves you on a road with a line in the middle.



As you approach a bend a sign on the right of the road advises you to slow down to the speed indicated. The main disappointment comes when you actually get to the bend. Instead of expecting to see the road curve round and having to go left or right to get round it, it is straight and the only thing that tells you that there is a bend is a couple of funny shaped signs on the right hand side of the road. If you go round a bend too fast you will either crash, spin off or just receive a warning. By the way, you only get one life!

Twice during a lap you are informed of the weather. If there is rain your braking power is reduced. There are only three kinds of weather "dry", "light rain", or "heavy rain".

You can also make pit stops. To do this you must come to a halt on the chequered part of the road. When you do this you will be refuelled and will be told about any faults that have been cleared. I advise you to make at least one pit stop during the race because on the two times I have got to the last lap my braking has failed and my accelerator jammed. The race lasts for five laps and believe me thats enough for anyone.

Another problem you have to face is mastering the keys, as there are ten commands you need to use. At the end of the game you are told the position you were in when you crashed and asked whether you would like to see the instructions, have the same track again, or have a new one generated.

For £5.45 (including postage) it is an excellent game for anyone with a 16K ZX81. I highly recommend it.

Ramlok Kit Adapt Electroni A.R. Parkins

The Ramlok Kit is designed to stop a ramlock from wobbling and causing program loss, a well known problem to Sinclair users. The main problem is that the edge connector does not provide an effective mechanical fixing to the back of the computer. Also the surface of the male connector at the edge of the computer is made of unreliable tinplate. This is what the kit aims to cut out. It consists of a connector and a mounting bracket.

The ramlok is fitted by removing the bottom of the computer, then placing the connector over and in line with the existing edge connector and soldering the underside connections. Then the circuit board is unscrewed and the upperside connections soldered. The board and bottom of the computer is then replaced. Next, the mounting bracket is placed on the computer, then aligned with rampack and stuck on. Two difference spacers are provided to account for all types of rampack. The rampack can be removed by pulling out a split pin in the mounting bracket. It can also be used on the ZX80 but due to different construction it is fitted differently.

I have found in use that the kit does provide successful non-wobble connectors. Although fixing on can be a bit tricky to people unknown to electronic assembly work. A spacer kit is also needed if to be used with a printer.

The kit costs £7.50 plus 50p. P&P from Adapt Electronic, 20 Starling Close, Buckhurst Hill, Essex IG9 5TN.

Road Up

Pit your wits against this holey program from Mike Bell in Norwich.

So you've collected your winnings and started for home; of course it would be just your luck for the bag to burst and the wind to blow your cash ($) all over the road. It would also just happen to be a road which was undergoing major works and is pitted with holes.

Using the cursor keys you can move your alter ego (*) around the maze-like streets and get back all twenty dollars. These are visible unlike the holes which have been covered over by the over-enthusiastic council. The outline of the street maze remains the same for each game, but the positions of the dollars and the holes are generated at random. Your score is constantly displayed and updated.

At the beginning of the game, the position of the holes and dollars are shown but one by one the holes are covered over, leaving you to remember their positions. Falling into one needless to say, is fatal.

```
  10 GOSUB 2000
  15 LET D=1
  16 LET A=0
  17 LET S=0
  20 PRINT AT D,A;"*";AT 0,0;"YO
UR SCORE=";S
  25 LET G$=INKEY$
  27 IF G$="" THEN GOTO 25
  30 PRINT AT D,A;" "
  40 LET A=A-(G$="5" AND A>0)+(G
$="8" AND A<31)
  50 LET D=D-(G$="7" AND D>0)+(G
$="6" AND D<31)
  60 IF A$(D+1,A+1)<>CHR$ 128 TH
EN GOTO 90
  70 LET A=A+(G$="5")-(G$="8")
  80 LET D=D+(G$="7")-(G$="6")
  90 IF A$(D+1,A+1)<>"$" THEN GO
TO 120
 100 LET A$(D+1,A+1)=" "
 110 LET S=S+1
 120 IF A$(D+1,A+1)<>"H" THEN GO
TO 20
1000 PRINT AT 21,0;"OH-YOU FELL
INTO A HOLE AND DIED"
1010 IF INKEY$="" THEN GOTO 1010
1020 RUN
1999 REM ***********************
         * Change all # into   *
         * inverse spaces in   *
         * following lines.    *
         ***********************
2000 DIM A$(22,32)
2010 LET A$(1)="
                "
2020 LET A$(2)=A$(1)
2030 LET A$(3)="  ###### #####
 ##### ######  "
2040 LET A$(4)="  #
             #  "
2050 LET A$(5)="  #    ####### #
 ####### ### #  "
2060 LET A$(6)="  #  # #       #
 #        # # #  "
2070 LET A$(7)="  #### # #######
 # ##### #   #  "
2080 LET A$(8)="  #  # #      #
 # #   # ##  #  "
2090 LET A$(9)="  #  # ##### # #
## # # #  #  #  "
2100 LET A$(10)="  #  #        #
#  ## # ## ####  "
2110 LET A$(11)="     ##### ###
#     #      #  "
2120 LET A$(12)="  #       #
 # ### #####    "
2130 LET A$(13)=" #### ## # ##
 # #        #  #  "
2140 LET A$(14)="  #  #  # # # #
## # ##### #  #  "
2150 LET A$(15)="  #  ## #    # #
   #      # ####  "
2160 LET A$(16)="  #    # ##### #
 ####### # #  #  "
2170 LET A$(17)="  # # #        #
 #        # #  #  "
2180 LET A$(18)="  # ### ########
 # ########    #  "
2190 LET A$(19)=A$(4)
2200 LET A$(20)=A$(3)
2210 LET A$(21)=A$(1)
2220 LET A$(22)=A$(21)
2260 FOR N=1 TO 20
2270 GOSUB 2340
2280 LET A$(D,A)="$"
2290 IF N/2<>INT (N/2) THEN GOTO
 2320
2300 GOSUB 2340
2310 LET A$(D,A)="H"
2320 NEXT N
2330 GOTO 2400
2340 LET A=INT (RND*28)+3
2350 LET D=INT (RND*18)+3
2360 IF A$(D,A)<>" " THEN GOTO 2
340
2370 RETURN
2400 FOR N=1 TO 22
2410 PRINT A$(N)
2420 NEXT N
2430 FOR Y=3 TO 20
2440 FOR Z=1 TO 32
2450 IF A$(Y,Z)="H" THEN PRINT A
T Y-1,Z-1;" "
2460 NEXT Z
2470 NEXT Y
2480 RETURN
```


A SENSATIONAL ADVANCE IN
COMPUTER GAME TECHNOLOGY
MICRO
COMMAND
YOU SPEAK–
THE COMPUTER OBEYS
Easily programmed voice control
Sensitive to individual voice prints
Instant response on screen
Your attention can stay on the screen – not on the controls
FOR USE WITH sinclair
ZX Spectrum
HOME COMPUTER UNITS
MAJOR PRICE BREAKTHROUGH
ONLY
£49.95 R.R.P.
Further Distributor, Dealer and Export Enquiries Welcome
ORION DATA
3 Cavendish Street, Brighton, East Sussex
BN2 1RN
Tel: (0273) 672994

T 1984

Cross the

C Stone from Hull poses the classic problem —

One day, a farmer was carrying a bag of corn home, accompanied by a hen and a wolf, when he came to a river. After a while the farmer found he could wade across the river carrying either the corn bag, the hen or the wolf. He then realised that if the hen and the corn were left together without his supervision, the hen would eat the corn. He also realised that if the wolf and hen were left unsupervised, the wolf would eat the hen. How did the farmer solve his problem?

The farmer can be made to carry one article across the river by pressing W (to transfer the wolf), H (to transfer the hen) or C (to transfer the corn bag). Remember that an article can only be transferred across the river if it is on the same bank as the farmer.

The program

Each graphic representation (farmer, corn bag, hen, wolf) is made up of four user-defined graphics characters. The procedure for defining these characters consists of lines 10 to 332. Lines 400 to 450 draw the bank and the river. Each character is given a numerical variable which is either 1 or −1 according to which bank they are on. This variable is then used to print the characters on the appropriate side of the bank.

The subroutine beginning at line 1000 ensures that any mistakes (e.g. wolf eats hen) are spotted and acts accordingly. If a mistake has been made, the player is informed and the game starts again. Line 1000 checks to see whether the puzzle has been successfully completed and, if this is the case, prints a message of congratultion. Line 520 gets a key-press from the keyboard and checks to see if it is either H, C, W or F (upper or lower case). No other input is permitted. Having verified that the key-press is O.K., lines 545 to 557 check whether the *move* is O.K. No illegal moves (e.g. pressing H when farmer and hen are on opposite banks) are allowed. Assuming all is well, lines 580 to 620 erase the present image of the characters to be moved, and also that of the farmer as he is always moved from bank to bank whether he carried something with him or not. When a character crosses the river, its assigned variable changes sign and this is accomplished via lines 630 and 550 to 570.

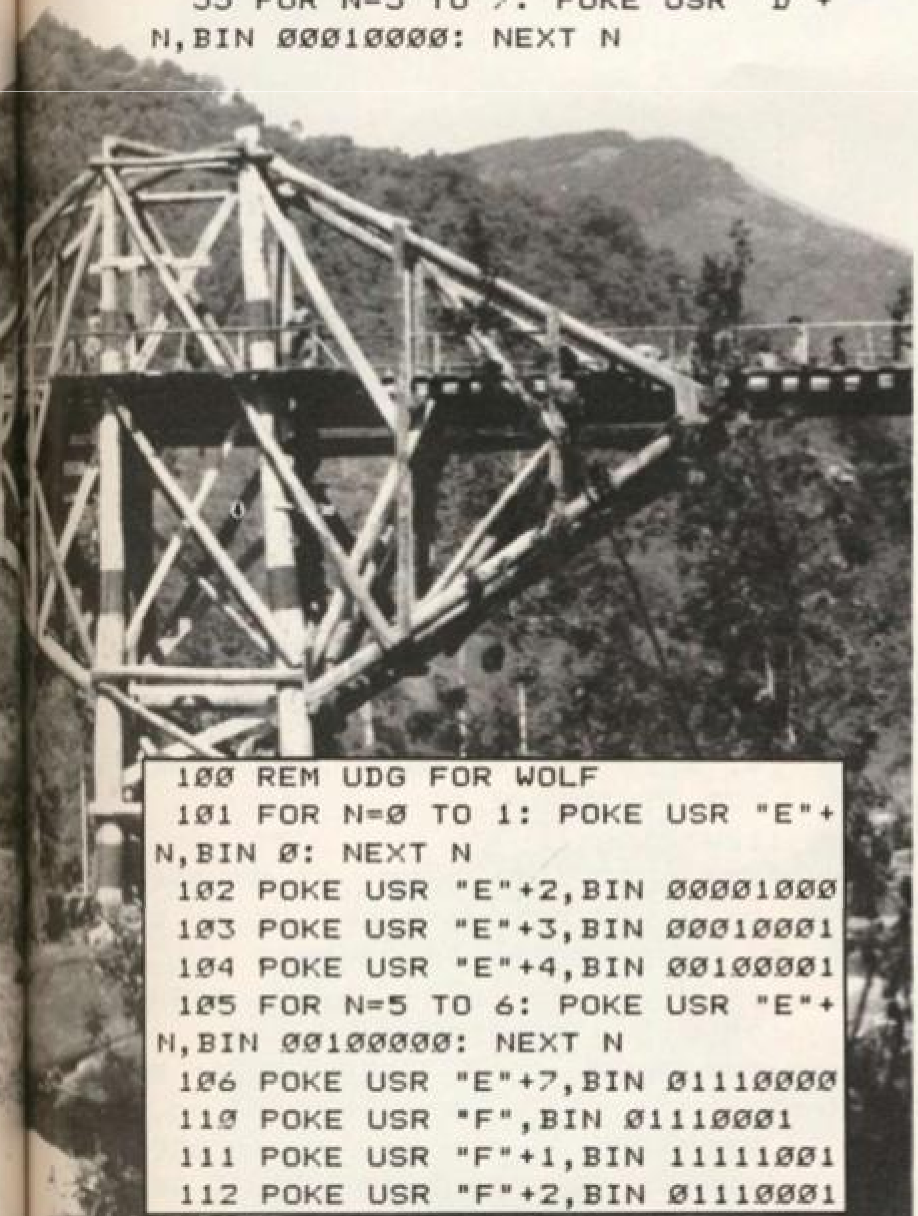


```
   1 REM CROSS THE RIVER
   2 REM BY CHRIS HOWARD STONE
   3 REM 8 BOULTON GROVE,HULL,HU
MBERSIDE,HU9 3ED
   4 REM MARCH 1983
  10 REM UDG FOR HEN
  20 POKE USR "A",BIN 01100000
  21 POKE USR "A"+1,BIN 11110000
  22 FOR N=2 TO 4: POKE USR "A"+
N,BIN 11111000: NEXT N
  23 POKE USR "A"+5,BIN 0111111
  24 POKE USR "A"+6,BIN 0011111
  25 POKE USR "A"+7,BIN 0000111
  30 POKE USR "B",BIN 00001111
  31 POKE USR "B"+1,BIN 0000011
  32 POKE USR "B"+2,BIN 0000000
  33 POKE USR "B"+3,BIN 0000001
  34 POKE USR "B"+4,BIN 0000010
```

River

Can YOU cross the river?

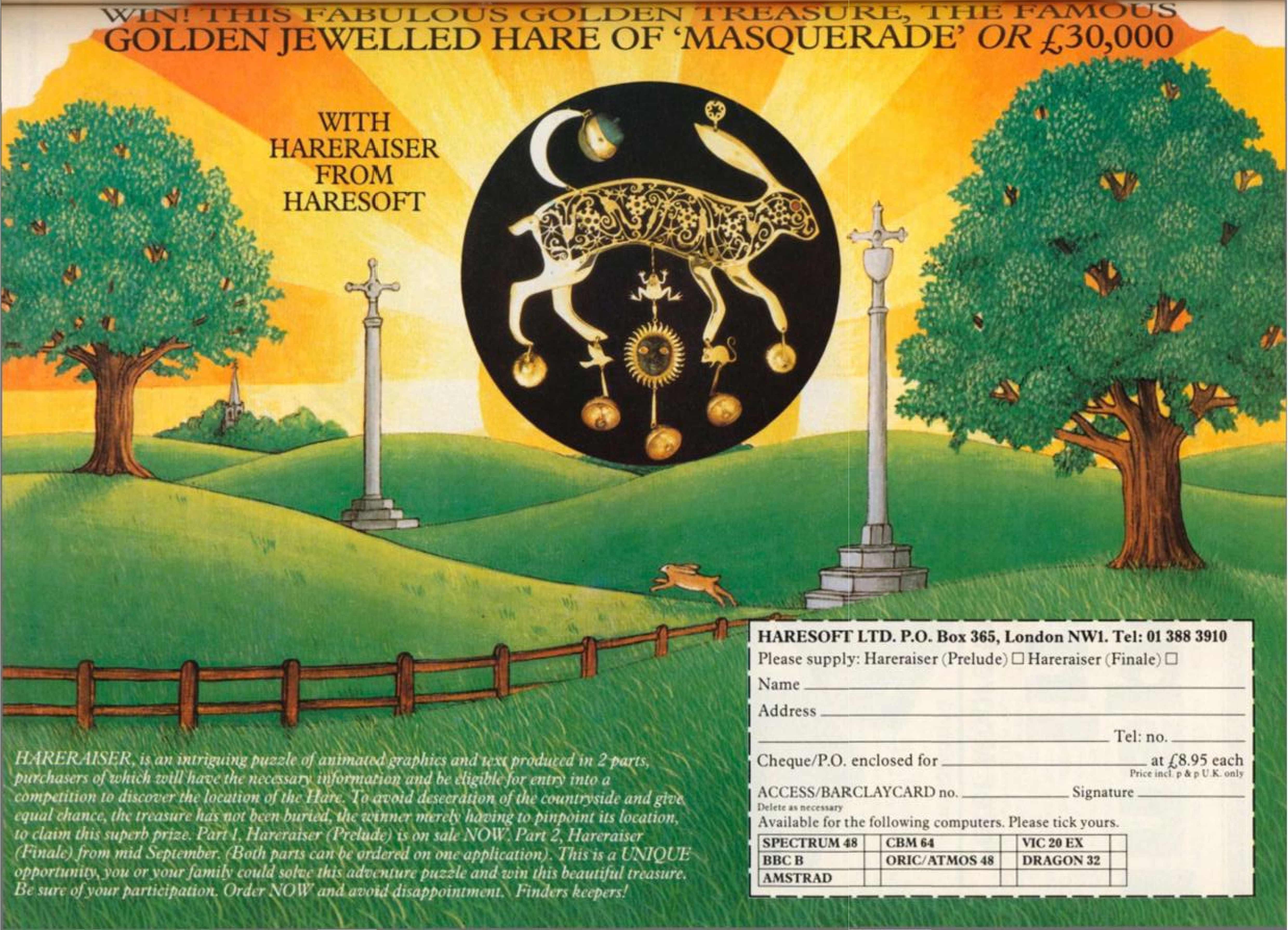
```
 35 POKE USR "B"+5,BIN ØØØØ1ØØØ
 36 POKE USR "B"+6,BIN ØØØØØ1ØØ
 37 POKE USR "B"+7,BIN ØØØØØØ1Ø
 4Ø POKE USR "C",BIN ØØØØ111Ø
 41 POKE USR "C"+1,BIN ØØØØ1111
 42 POKE USR "C"+2,BIN ØØØØ111Ø
 43 POKE USR "C"+3,BIN ØØØØ11ØØ
 44 POKE USR "C"+4,BIN ØØØ11ØØØ
 45 POKE USR "C"+5,BIN 1111ØØØØ
 46 POKE USR "C"+6,BIN 111ØØØØØ
 47 POKE USR "C"+7,BIN 1111ØØØØ
 5Ø POKE USR "D",BIN 111ØØØØØ
 51 POKE USR "D"+1,BIN 11ØØØØØØ
 52 POKE USR "D"+2,BIN 1ØØØØØØØ
 53 POKE USR "D"+3,BIN Ø1ØØØØØØ
 54 POKE USR "D"+4,BIN ØØ1ØØØØØ
 55 FOR N=5 TO 7: POKE USR "D"+
N,BIN ØØ1ØØØØØ: NEXT N
```



```
 1ØØ REM UDG FOR WOLF
 1Ø1 FOR N=Ø TO 1: POKE USR "E"+
N,BIN Ø: NEXT N
 1Ø2 POKE USR "E"+2,BIN ØØØØ1ØØØ
 1Ø3 POKE USR "E"+3,BIN ØØØ1ØØØ1
 1Ø4 POKE USR "E"+4,BIN ØØ1ØØØØ1
 1Ø5 FOR N=5 TO 6: POKE USR "E"+
N,BIN ØØ1ØØØØØ: NEXT N
 1Ø6 POKE USR "E"+7,BIN Ø111ØØØØ
 11Ø POKE USR "F",BIN Ø111ØØØ1
 111 POKE USR "F"+1,BIN 11111ØØ1
 112 POKE USR "F"+2,BIN Ø111ØØØ1
 113 POKE USR "F"+3,BIN Ø111ØØØØ
 114 POKE USR "F"+4,BIN ØØ11ØØ11
 115 POKE USR "F"+5,BIN ØØ111111
 116 POKE USR "F"+6,BIN ØØØ111ØØ
 117 POKE USR "F"+7,BIN Ø
 12Ø FOR N=Ø TO 1: POKE USR "G"+
N,BIN 1ØØ1ØØØØ: NEXT N
 121 POKE USR "G"+2,BIN 1111ØØØØ
 122 POKE USR "G"+3,BIN 1ØØ11ØØØ
 123 POKE USR "G"+4,BIN 11111ØØØ
 124 POKE USR "G"+5,BIN 1111ØØØØ
 125 POKE USR "G"+6,BIN Ø11ØØØØØ
 126 POKE USR "G"+7,BIN 1111ØØØØ
 13Ø FOR N=Ø TO 2: POKE USR "H"+
N,BIN 11111ØØØ: NEXT N
 131 FOR N=3 TO 4: POKE USR "H"+
N,BIN 1111ØØØØ: NEXT N
 132 FOR N=5 TO 7: POKE USR "H"+
N,BIN 1ØØ1ØØØØ: NEXT N
 2ØØ REM UDG FOR CORN BAG
 2Ø1 POKE USR "I",BIN Ø
 2Ø2 POKE USR "I"+1,BIN ØØØ1ØØØØ
 2Ø3 POKE USR "I"+2,BIN ØØØØ11ØØ
 2Ø4 POKE USR "I"+3,BIN ØØØØ1111
 2Ø5 FOR N=4 TO 7: POKE USR "I"+
N,BIN ØØØØØ111: NEXT N
 21Ø FOR N=Ø TO 3: POKE USR "J"+
N,BIN ØØØØØ111: NEXT N
 211 POKE USR "J"+4,BIN ØØØØ1111
 212 POKE USR "J"+5,BIN ØØØØ11ØØ
 213 POKE USR "J"+6,BIN ØØØ1ØØØØ
 214 POKE USR "J"+7,BIN Ø
 22Ø POKE USR "K",BIN Ø
 221 POKE USR "K"+1,BIN ØØØØ1ØØØ
 222 POKE USR "K"+2,BIN ØØ11ØØØØ
 223 POKE USR "K"+3,BIN 1111ØØØØ
 224 FOR N=4 TO 7: POKE USR "K"+
N,BIN 111ØØØØØ: NEXT N
 23Ø FOR N=Ø TO 3: POKE USR "L"+
N,BIN 111ØØØØØ: NEXT N
 231 POKE USR "L"+4,BIN 1111ØØØØ
 232 POKE USR "L"+5,BIN ØØ11ØØØØ
 233 POKE USR "L"+6,BIN ØØØØ1ØØØ
 234 POKE USR "L"+7,BIN Ø
 3ØØ REM UDG FOR FARMER
 3Ø1 FOR N=Ø TO 2: POKE USR "M"+
N,BIN ØØØØØØ11: NEXT N
 3Ø2 POKE USR "M"+3,BIN ØØØØØØØ1
 3Ø3 FOR N=4 TO 5: POKE USR "M"+
N,BIN ØØØ11111: NEXT N
 3Ø4 FOR N=6 TO 7: POKE USR "M"+
N,BIN ØØØ1Ø111: NEXT N
 31Ø FOR N=Ø TO 1: POKE USR "N"+
N,BIN ØØØ1Ø111: NEXT N
 311 POKE USR "N"+2,BIN ØØØØØ111
 312 FOR N=3 TO 7: POKE USR "N"+
N,BIN ØØØØØØ1Ø: NEXT N
 32Ø FOR N=Ø TO 2: POKE USR "O"+
N,BIN 11ØØØØØØ: NEXT N
```

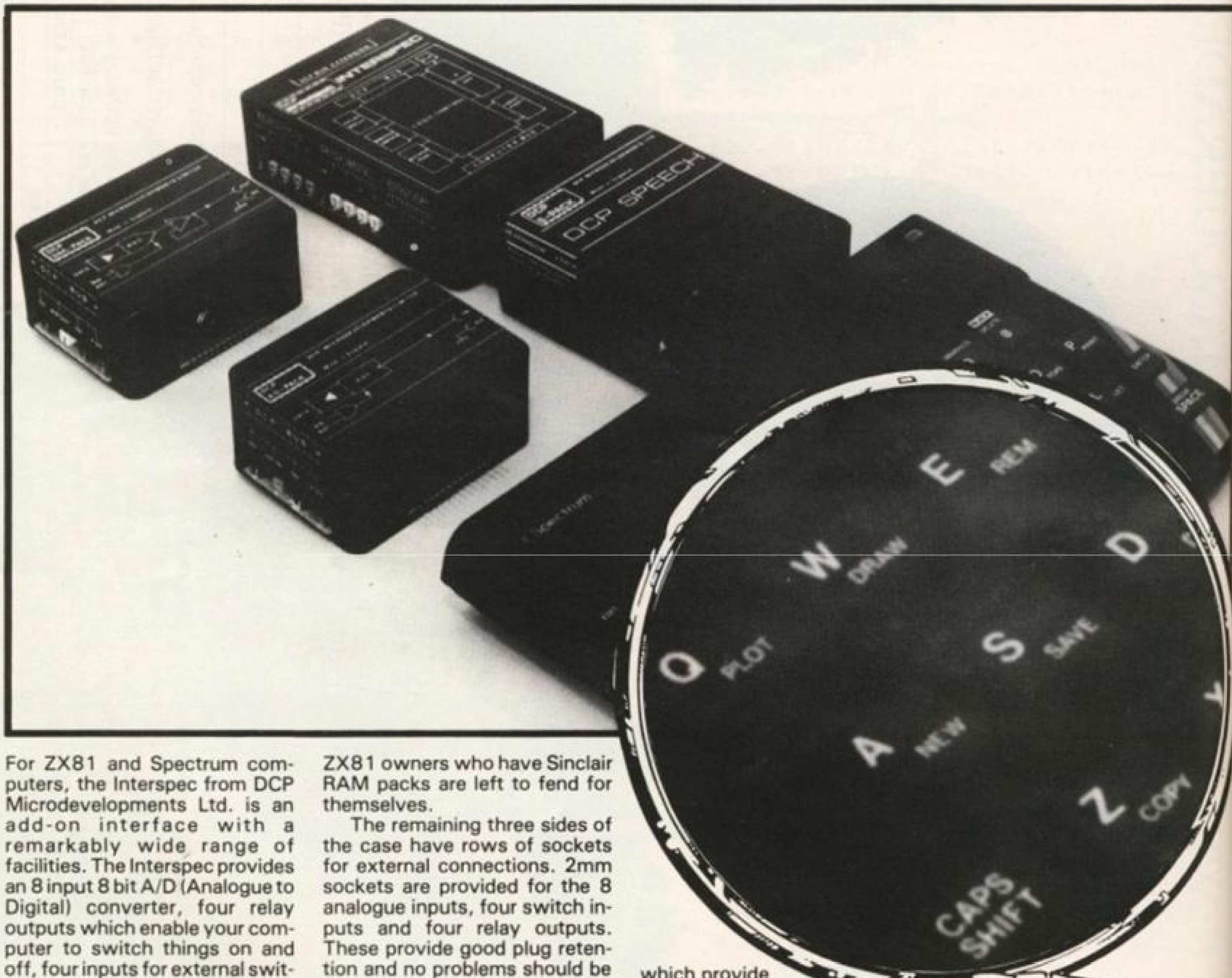
```
 321 POKE USR "O"+3,BIN 1ØØØØØØ
 322 FOR N=4 TO 5: POKE USR "O"+
N,BIN 1111ØØØ: NEXT N
 323 FOR N=6 TO 7: POKE USR "O"+
N,BIN 111Ø1ØØØ: NEXT N
 33Ø FOR N=Ø TO 1: POKE USR "P"+
N,BIN 111Ø1ØØØ: NEXT N
 331 POKE USR "P"+2,BIN 11ØØØØØ
 332 FOR N=3 TO 7: POKE USR "P"+
N,BIN Ø1ØØØØØ: NEXT N
 4ØØ>REM DRAW RIVER AND BANK
 41Ø FLASH Ø: BORDER Ø: PAPER 4:
 CLS
 42Ø PAPER 7: FOR A=Ø TO 4: PRIN
T AT A,14;"      ": NEXT A: FOR A=
18 TO 21: PRINT AT A,14;"      ":
NEXT A
 43Ø FOR A=5 TO 7: PRINT AT A,8;
"                ": NEXT A: FOR
A=15 TO 17: PRINT AT A,8;"
          ": NEXT A
 44Ø FOR B=1 TO 7: PRINT  PAPER
5;AT B+7,Ø;"
             ": NEXT B
 45Ø LET f=1: LET h=1: LET c=1:
LET w=1
 46Ø REM MAIN LOOP
 465 REM * Characters in lines *
         * 48Ø to 51Ø are UDG's*
 47Ø INK Ø
 48Ø PRINT AT 6*(f=1)+15*(f=-1),
9;"M";: PRINT "O": PRINT AT 7*(f
=1)+16*(f=-1),9;"N";: PRINT "P"
 49Ø PRINT AT 6*(h=1)+15*(h=-1),
13;"A";: PRINT "C": PRINT AT 7*(
h=1)+16*(h=-1),13;"B";: PRINT "D
"
 5ØØ PRINT AT 6*(c=1)+15*(c=-1),
17;"I";: PRINT "K": PRINT AT 7*(
c=1)+16*(c=-1),17;"J";: PRINT "L
"
 51Ø PRINT AT 6*(w=1)+15*(w=-1),
21;"E";: PRINT "G": PRINT AT 7*(
w=1)+16*(w=-1),21;"F";: PRINT "H
": GO SUB 1ØØØ
 52Ø IF INKEY$="" THEN  GO TO 52
Ø
 53Ø LET m$=INKEY$: IF m$="h" OR
 m$="H" OR m$="c" OR m$="C" OR m
$="w" OR m$="W" OR m$="f" OR m$=
"F" THEN  GO TO 545
 54Ø BEEP 1,-12: GO TO 52Ø
 545 IF (m$="c" OR m$="C") AND f
*c<>1 THEN  GO TO 54Ø
 546 IF (m$="h" OR m$="H") AND h
*f<>1 THEN  GO TO 54Ø
 547 IF (m$="w" OR m$="W") AND w
*f<>1 THEN  GO TO 54Ø
 55Ø BEEP .4,24: IF m$="h" OR m$
="H" THEN  LET h=-1*h
 56Ø IF m$="c" OR m$="C" THEN  L
ET c=-1*c
 57Ø IF m$="w" OR m$="W" THEN  L
ET w=-1*w
 58Ø PRINT AT 6*(f=-1)+15*(f=1),
9;"  ": PRINT AT 7*(f=-1)+16*(f=
1),9;"  "
 59Ø PRINT AT 6*(h=-1)+15*(h=1),
13;"  ": PRINT AT 7*(h=-1)+16*(h
=1),13;"  "
 6ØØ PRINT AT 6*(c=-1)+15*(c=1),
17;"  ": PRINT AT 7*(c=-1)+16*(c
=1),17;"  "
 61Ø PRINT AT 6*(w=-t)+15*(w=1),
21;"  ": PRINT AT 7*(w=-1)+16*(w
=1),21;"  "
 62Ø PRINT AT 6*(f=1)+15*(f=-1),
9;"  ": PRINT AT 7*(f=1)+16*(f=-
1),9;"  "
 63Ø LET f=-1*f
 64Ø GO TO 46Ø
1ØØØ IF F=-1 AND H=-1 AND C=-1 A
ND W=-1 THEN  GO TO 1Ø8Ø
1Ø1Ø REM ERROR DETECTION
1Ø2Ø IF SGN H<>SGN C THEN  GO TO
 1Ø5Ø
1Ø3Ø IF SGN F=SGN C THEN  GO TO
1Ø5Ø
1Ø4Ø PAPER 2: INK 6: FLASH 1: PR
INT AT 2,9;"HEN EATS CORN": GO T
O 13ØØ
1Ø5Ø IF SGN W<>SGN H THEN  RETUR
N
1Ø6Ø IF SGN W=SGN F THEN  RETURN

1Ø7Ø PAPER 2: INK 6: FLASH 1: PR
INT AT 2,9;"WOLF EATS HEN": GO T
O 12ØØ
1Ø8Ø PAPER 3: INK 6: FLASH 1: PR
INT AT 1,8;"CONGRATULATIONS": PR
INT AT 2,3;"YOU HAVE CROSSED THE
 RIVER"
1Ø9Ø BEEP .25,5: BEEP .25,5: BEE
P .25,5: BEEP .25,Ø: BEEP .25,2:
 BEEP .25,2: BEEP .5,Ø: BEEP .25
,9: BEEP .25,9: BEEP .25,7: BEEP
 .25,7: BEEP 1,5
11ØØ STOP
12ØØ BEEP 1,Ø: BEEP .75,Ø: BEEP
.25,Ø: BEEP 1,Ø: BEEP .75,4: BEE
P .25,2: BEEP .75,2: BEEP .25,Ø:
 BEEP .75,Ø: BEEP .25,-1: BEEP 1
.25,Ø: GO TO 4ØØ
13ØØ BEEP .25,Ø: BEEP .25,4: BEE
P .25,Ø: BEEP .25,4: BEEP .25,Ø:
 BEEP .75,4: FOR M=1 TO 4: BEEP
.125,4: NEXT M: BEEP .25,2: BEEP
 .375,2: BEEP 1,Ø: GO TO 4ØØ
```


WIN! THIS FABULOUS GOLDEN TREASURE, THE FAMOUS
GOLDEN JEWELLED HARE OF 'MASQUERADE' OR £30,000
WITH
HARERAISER
FROM
HARESOFT
HARERAISER, is an intriguing puzzle of animated graphics and text produced in 2 parts, purchasers of which will have the necessary information and be eligible for entry into a competition to discover the location of the Hare. To avoid desecration of the countryside and give equal chance, the treasure has not been buried, the winner merely having to pinpoint its location, to claim this superb prize. Part 1, Hareraiser (Prelude) is on sale NOW. Part 2, Hareraiser (Finale) from mid September. (Both parts can be ordered on one application). This is a UNIQUE opportunity, you or your family could solve this adventure puzzle and win this beautiful treasure. Be sure of your participation. Order NOW and avoid disappointment. Finders keepers!
HARESOFT LTD. P.O. Box 365, London NW1. Tel: 01 388 3910
Please supply: Hareraiser (Prelude) ☐ Hareraiser (Finale) ☐
Name
Address
Tel: no.
Cheque/P.O. enclosed for
at £8.95 each
Price incl. p & p U.K. only
ACCESS/BARCLAYCARD no.
Signature
Delete as necessary
Available for the following computers. Please tick yours.
SPECTRUM 48
CBM 64
VIC 20 EX
BBC B
ORIC/ATMOS 48
DRAGON 32
AMSTRAD

Inspecting the Interspec

Peter Moore looks at a unit to put you in control.



For ZX81 and Spectrum computers, the Interspec from DCP Microdevelopments Ltd. is an add-on interface with a remarkably wide range of facilities. The Interspec provides an 8 input 8 bit A/D (Analogue to Digital) converter, four relay outputs which enable your computer to switch things on and off, four inputs for external switches, an 8 bit input port, 8 bit output port and a facility for further expansion through a 15 way connector called the DCP Bus.

The Interspec is housed in the familiar black plastic case used by so many manufacturers. DCP have worked wonders with this case! It really looks good. The unit lies flat on the desk with the edge connector that plugs into your computer at one end of the case. No rear edge connector is provided on the Interspec so ZX81 owners who have Sinclair RAM packs are left to fend for themselves.

The remaining three sides of the case have rows of sockets for external connections. 2mm sockets are provided for the 8 analogue inputs, four switch inputs and four relay outputs. These provide good plug retention and no problems should be experienced with wires coming unplugged. 0.1" 'Molex' connectors are provided for the 8 bit input and output ports and the DCP Bus.

The Interspec electronics are mounted on a double-sided printed circuit board and the component layout is neat and professional.

The instruction booklet gives full details on programming the Spectrum and a supplementary leaflet contains two short machine code subroutines which provide the equivalent of IN and OUT commands on the ZX81.

A to D Interface

An A/D (Analogue to Digital) interface converts a voltage at an input to a number in the computer. Eight analogue inputs are provided on the Interspec which can be used for signals in the range 0 to 2.7 volts. DCP quote a tolerance of +/− 20%

although on the unit I tested, accuracy was well within 10% in the range 0 to 2 volts. The interface has a resolution of 10 millivolts (10mV); if you were to put a signal of 10mV in, your computer would read this as 1: 500mV would read as 50, while 1 volt would be 100.

Programming the A/D interface is easy. Each of the 8 analogue inputs has a number from 0 to 7. To select one of these inputs, the command OUT 31,n is used where n is a number from 0 to 7: then LET x = IN 31 reads the numerical equivalent of the voltage on the selected input to variable x. The OUT command must be repeated before each IN since this initialises A/D conversion. The Interspec uses the AD0809 A/D chip; DCP quote the conversion time as being typically better than 1 millisecond.

While anyone familiar with electronics theory knows how a graph showing the charge and discharge curves of a capacitor will appear, seeing it happen in real time with the computer drawing a graph on the screen is a fascinating demonstration of the theory.

I must admit to spending some time charging and discharging capacitors and watching the varying voltages across them being plotted by my Spectrum. A simple D/A (Digital to Analogue) circuit using the ZN428 is described in the Interspec literature, details of the connections to be made to the Interspec DCP Bus are included. Personally I would have liked to see this built into the Interspec — perhaps in place of one of the A/D inputs; however the extra circuitry is very straightforward and should not require much experience to build.

Relay Outputs

Four subminiature relays are built into the Interspec. These have contacts rated 1 amp at 12 volts. Each relay output has a binary value of 1, 2, 4 or 8. Alongside these outputs is a socket marked 'relay common' to which are connected one half of each of the four pairs of relay contacts. These four outputs can be used to directly switch small motors, solenoids, lamps etc. If you want to switch mains powered equipment on and off, you will have to use the Interspec relays to switch larger 'heavier' relays whose contacts are used to switch the mains. Again, programming is by means of IN and OUT commands. OUT 63,4 would switch relay four's contacts on; OUT 63,0 would switch all four relays off.

Four inputs for switches are provided which give an easy means of connecting external switches to your computer. A fifth socket supplying +5 volts is mounted next to these. Switches should be wired so that they connect this 5 volt line to a switch input. Programming for the switch input port takes the form LET X = (IN 63) − 240. Each of the four inputs has a binary value of 1, 2, 4 or 8. So if switches 1 and 4 were closed, for instance, the variable x would return the value 5 (i.e. 4 + 1 = 5).

Eight bit ports

The 8 bit input and output ports are programmed with IN and OUT 95. IN 95 reads in a byte from the 8 bit input port, while OUT 95 outputs a byte of data (i.e. a number from 0 to 255) to the 8 bit output port. Data is latched to the output port and remains constant until changed, so this port could be used, for example, to light LEDs.

DCP have also devised a simple 15 way connection system which allows you to expand your computer's capabilities with further electronics. DCP market other equipment which plugs into this DCP Bus including a D/A converter and a fast A/D converter. Both these devices cost £19.95 + £1.95 postage.

Conclusion

The DCP Interspec is undoubtedly a very useful tool for educational purposes and should prove an interesting accessory for computer and electronics hobbyists. At £49.95 + £1.95 postage it is perhaps a little expensive for the average micro user but you get a lot for your money.

The Interspec can be obtained directly from DCP Microdevelopments Ltd., at 2, Station Close, Lingwood, Norwich.

Mindplay



Intrepid Adventurer Greg Turnbull continues his series of reviews for lovers of 'arcade-alternatives'

Pacific War
Cases Computer Simulations Ltd.
Price £5.95

PACIFIC WAR is a wargame simulation of the battle of Guadalcanal in the Eastern Solomons, August 1942. Only one player is allowed; he controls the American, and the computer the Japanese, forces. The objective is to find, and destroy the enemy's fleets, before they land reinforcements, or destroy you. The format is similar to the Lothlorien games; but uses slightly better graphics. Play is shown on a grid-map of the area (the references are strangely double letters, e.g. KS?).

There are nine levels of play, although there isn't much apparent difference between each one. A screen dump of the initial map will prove very useful during the game. Play consists of setting up your task-forces, selecting (alterable) courses; then sequential game turns (G.T.) of three operations turns (O.T.) each. You control three task-forces; battleships, cruisers, aircraft-carriers and transporters; plus Henderson air field. The computer has six task-forces (doesn't it always have the advantage?)

Progress is made on a points system after 'Battleships', on which this game is based. Set numbers of hits being required to sink each ship. The scores are totalled, and the first to reach a set value wins the 'battle'. Half of this large program is in machine code; although play is often slow and very repetitive, eg. target movement by cursor keys, form the lower border, one square per time! (three would have been better.)

Each task-force may have up to 3 air missions (which move 2 squares per O.T.), and only move one square per G.T. Early play entails sending numerous search missions to locate the enemy. Once found, airstrikes (of fighters, torpedos, or dive-bombers) can be sent out, to try and destroy the opposing task-force. Previous sightings are shown on the map, for a few turns, by an expanding square. The only defence against Japanese air attacks is to launch a C.A.P. (combat aircraft patrol), which decreases your fighter strength. Strong air strikes are most effective, but once sent they cannot be recalled.

Status reports on each task-force are available, and after each G.T. a full fleet damage report is given. Note-making and tactics are all-important. This is a long game, and inexplicably no 'SAVE/LOAD' option is provided! The rules are complex but easily learnt, but the game becomes somewhat boring due to constant repetition, and slowness. The only outstanding feature being the moving-graphic 'planes on the map.

Each game is varied because of the random start positions, but soon become the same as the last (despite the level chosen). To re-play the game you are required to re-load the entire program! In all, an interesting strategy game; only *just* recommended. It could have been so much better with just a little more careful thought, and variety in play.

Castle Blackstar
SCR Adventures

This is the first in a series of 'Artemis quests' from **SCR Adventures**. It is a text-only, machine code program giving extremely fast response times. Essentially it is a medieval fantasy, similar in theme to 'Knight's Quest'. The object is to enter the castle and recover the power orb, located in the vast (no understatement) underground caves.

Maximum (240) points are awarded for collecting all the treasures, completing certain special tasks (by overcoming puzzles), and returning the orb to the Goddess. All the usual commands are understood from a comprehensive vocabulary of some 200 words (it uses approx. 40K of memory.) If you get frustrated at any point, don't be tempted to swear, as this will cost you one object! There are many nice, humorous touches along the way, as well as a few red-herrings.

The game is not in real-time, so you can sit back and think your way out of each situation. It isn't interactive either, so you can't go round talking to the few characters you occasionally meet. Single and four-letter strings are used. For exmaple: I = inventory, L = look, SCOR = points so far, N, S, SW, NE, U, D, etc. Plus some new and interesting commands: 'GOES' for number of turns taken, 'VERBOSE' for full descriptions, and 'QUIET' for shorter text on each location. (Although I couldn't see that 'quiet' had any noticeable effect!) This feature is not usually seen on anything smaller than a disc-based program.

The description themselves are very long (they may even fill an entire screen), and certainly create the atmosphere appropriate to the game's period. Sadly there is no provision for a 'HELP' feature for those of us who get stuck easily. Thankfully 'QUIT', 'SAVE', and 'LOAD' are all supported. One feature I do like is that there are a number of possible ways around some of the problems. This means that you don't suddenly come to an impasse (akin to hitting a brick wall), as often happens in some other adventures.

One awkward point is that not all the available directions are displayed on the screen — you have to guess — and it's easy to forget to try all of the possible ways. There are **many** locations, so good mapping is essential (especially as there are numerous levels: castle; upper and lower levels, two towers, tunnels and caverns, plus the huge cave system.) Fortunately these are very accessible, so without having to solve too many problems you can easily get to see over 90 locations. This is very useful for the novice. However the puzzles are challenging enough to keep the hardened adventurer busy for quite some time (due to the fact that they were written by fellow adventure-player.)

Hidden in the game are a number of death-traps. Dark places are to be avoided for fear of very nasty Deodans, who appear to enjoy decapitating unwary players! Getting killed isn't necessarily the end of the game; you merely drop all objects at that location, lose some points, and very handily get reincarnated back at the start. Hint: Beware of the forest, without the objects needed to map it, and remember the lamps and oil can run out.

The program also contains some oblique references to various literary sources. For example: a pentagram and an Angel of Death from 'The Devil rides out', a sword in a stone from the 'Arthur' legends, and runes about Smaug on a Mithril knife from a certain well-known book! The manual is well-written, but doesn't give anything away. SCR can be contacted if you really get stuck; and if you're brilliant the first 50 to complete the quest get free 'Castle Blackstar' T-shirts. Overall an excellent adventure, especially for the beginner. The next one in this series, 'Pyramid of the sun' is due out later this year. If it's as good as this one, it should certainly be worth playing.

nately
ble, so
re too
easily
ations.
r the
les are
ep the
sy for
ne fact
fellow

are a
Dark
or fear
ho ap-
tating
killed
of the
all ob-
some
et rein-
. Hint:
ithout
ap it,
nd oil

ntains
es to
or ex-
d an
Devil
stone
, and
Mithril
nown
well-
give
e con-
tuck;
st 50
t free
hirts.
ture,
. The
ramid
r this
ne, it
play-

1984

Minotaur

For fans of 3D games, here is one of the best from regular contributor C C Stock of Chelmsford.

An evil king has for his amusement dropped you down a shaft into the Minotaurs maze. To make it more interesting you are unarmed, but possess a map showing you the positions of a sword, a helmet and a shield. Only when you have all of these can you kill the Minotaur. Once in the maze a total of 8 section scan be seen including the one you are in. Movement is very fast as the maze is only altered a little, and not totally redrawn. Thus you have "real time" motion. The Minotaur comes in 7 sizes, growing as he approaches you.

To see the map "M" is pressed. Whilst the map is shown the program is halted.

The maze is made from 4 strings, 2 for horizontal and 2 for vertical passages. "0" represents a break in the corridor, "5" represents the normal corridor, and "1"-"4" the passages that cross at that point. To redraw the maze change lines 60-140 and 2401, plus on line 20 x1 and y1 adjust to Minotaurs position plus 2.

Also, string numbers must alternate.

Summary

15-20	Introduction
60-80	The maze strings
100-180	Map picture
200-280	Draw corridor
300-350	Draw knight
400-530	Draw side passengers
610-630	Can Minotaur be seen?
700-770	Controls
800-840	Draw corridor end
850-870	Variables altered when turning
1000-1170	Minotaurs Movement
2000-2100	Draw Minotaur
2401	Draw map
3000-3030	Finish
3200-3350	RND positioning of armour and man
3400-3440	Pick up armour
4000-4540	Minotaur and man

```
   Ø> REM  ***** 3D MAZE *****
   @ C.C.Stock 1983
   1Ø INK Ø: PAPER 4: BORDER 2: C
LS
   15 PRINT  AT 6,7;" M I N O T A
 U R "; AT 8,Ø;"  For the amusem
ent of the king  you have been d
ropped into the  Minotaurs maze.
 You only hope   is to find 3 we
apons hidden in  the maze using
a map provided "
   2Ø GO SUB 4ØØØ
   3Ø DIM a$(4,34): LET a1=3: LET
 l=-1: LET j=-1: LET x1=2Ø: LET
y1=16: LET q1=1: LET s=1: LET t1
=Ø: LET t2=Ø: LET t3=Ø
   4Ø LET k=1: IF  RND <.5 THEN
LET k=-1
   6Ø LET a$(1)="Ø3545535540Ø3554
5553555545304555З0"
   7Ø LET a$(2)="Ø354553554553554
5553ØØØØ45354555ЗØ"
   75 LET a$(3)="Ø155255510Ø25551
Ø255152Ø"
   8Ø LET a$(4)="Ø1552ØØØ15525551
52ØØ152Ø"
  1Ø1 LET e$="----------██---------
-------█----▾"
  11Ø LET f$="------------------
████----------"
  12Ø LET g$="-█-██-██-██-██-███-
████-█-█-███-"
  13Ø LET h$="-████-█████-██████-
██████-█████-"
  14Ø LET i$="██-█████-█████-████
████-███-████"
  145 PRINT  AT 18,4;"Use keys 5,
6,7&8 to move"; AT 2Ø,8;"Press a
ny key"
  15Ø IF  INKEY$ ="" THEN  GO TO
15Ø
  16Ø GO SUB 24ØØ: GO SUB 32ØØ
  17Ø PRINT  AT 8,16; INVERSE 1;"
Press"; AT 9,16;"any key": INK Ø
: PAPER 6: FOR n=1 TO 3: PRINT
AT w(1,n),w(2,n);"X": NEXT n
  18Ø BEEP .8,-16: BEEP .8,-22: I
F  INKEY$ ="" THEN  GO TO 18Ø
  19Ø INK 6: PAPER 2: CLS : PAPER
 Ø
  195 PRINT  AT 3,17;"Press M for
 map"; AT 5,17;"Press A to pick"
; AT 6,17;" up armour "
  2Ø1 FOR n=Ø TO 21: PRINT  AT n,
Ø;"                ": NEXT n
  2Ø5 LET m=8
  21Ø FOR n=Ø TO m-1
  23Ø IF a$(a,n*k+p)="Ø" THEN  LE
T m=n+1: GO TO 3ØØ
  26Ø PLOT 8*n,16*n: DRAW 8,16: D
RAW Ø,159-16*n
  27Ø PLOT 127-8*n,16*n: DRAW -8,
16: DRAW Ø,159-16*n
  28Ø NEXT n
  31Ø FOR n=Ø TO 6
```






```
 32Ø PRINT  AT 14+n,6; INVERSE 1
;r$(1+4*n TO 4+4*n): NEXT n
 33Ø IF t1=1 THEN  PRINT  AT 15,
5; INK 6;" "; AT 16,5;" "; AT 17
,5;" "; AT 9,17; INVERSE 1;" Swo
rd  "
 34Ø IF t2=1 THEN  FOR n=Ø TO 3:
 PAPER 4: PRINT  AT 15+n,9; INK
Ø;"/": NEXT n: PRINT  AT 11,17;"
 Shield ": PAPER Ø: PLOT 72,48:
DRAW 7,Ø: PLOT 72,32: DRAW 7,Ø
 35Ø IF t3=1 THEN  PRINT  AT 14,
7; PAPER 2;"WW"; AT 1Ø,17;" Helm
et "
 41Ø PLOT Ø,Ø: DRAW 8,16: DRAW
INVERSE 1;-7,Ø: PLOT 127,Ø: DRAW
 -8,16: DRAW  INVERSE 1;7,Ø
 42Ø FOR n=m-1 TO Ø STEP -1
 43Ø LET b= VAL a$(a,p+n*k)
 44Ø IF b=5 THEN  GO TO 52Ø
 45Ø IF b=Ø THEN  LET m=m-1: GO
TO 8Ø1
 46Ø IF b=1 OR b=2 THEN  IF a$(b
,x+k)="Ø" THEN  GO TO 49Ø
 47Ø IF b=4 OR b=3 THEN  IF a$(b
,y-k)="Ø" THEN  GO TO 5Ø1
 48Ø PLOT 8*n,16*n: DRAW  INVERS
E 1;8,16: DRAW -7,Ø: DRAW 7,Ø: D
RAW 8,16: DRAW  INVERSE 1;-7,Ø
 49Ø IF b=1 OR b=2 THEN  IF a$(b
,x-k)="Ø" THEN  GO TO 52Ø
 5Ø1 IF b=3 OR b=4 THEN  IF a$(b
,y+k)="Ø" THEN  GO TO 52Ø
 51Ø PLOT 127-8*n,16*n: DRAW  IN
VERSE 1;-8,16: DRAW 7,Ø: DRAW -7
,Ø: DRAW -8,16: DRAW  INVERSE 1;
7,Ø
 52Ø NEXT n
 53Ø PLOT m*8,m*16: DRAW  INVERS
E 1;8,16: PLOT 127-m*8,m*16: DRA
W  INVERSE 1;-8,16
 6Ø1 PAUSE 3Ø: IF  RND <.7 THEN
 GO SUB 1ØØØ
 61Ø IF x=x1 AND y=y1 THEN  GO T
O 3ØØØ
 62Ø IF q=-1 THEN  LET y=p: IF x
=x1 AND (y1-y)*k <= m THEN  LET
d=(y1-y)*k: IF d>-1 THEN  GO SUB
 2Ø1Ø
 63Ø IF q=1 THEN  LET x=p: IF y=
y1 AND (x1-x)*k<m THEN  LET d=(x
1-x)*k: IF d>-1 THEN  GO SUB 2Ø1
Ø
 64Ø INK 6
 71Ø IF  INKEY$ ="a" THEN  BEEP
.1,24: GO TO 34ØØ
 72Ø IF  INKEY$ ="m" THEN  GO SU
B 24ØØ: GO TO 17Ø
 73Ø IF  INKEY$ ="6" THEN  LET k
=k*-1: GO TO 2Ø1
 74Ø IF  INKEY$ ="7" THEN  IF a$
(a,p+k) <> "Ø" THEN  LET p=p+k:
BEEP .2,Ø: GO TO 4Ø1
 75Ø IF  INKEY$ ="5" AND b <> 5
THEN  LET k=-q*k: GO SUB 85Ø: GO
 TO 2Ø1
 76Ø IF  INKEY$ ="8" AND b <> 5
THEN  LET k=q*k: GO SUB 85Ø: GO
TO 2Ø1
 77Ø GO TO 6Ø1
 81Ø PLOT 8*(n+1),175: DRAW  INV
ERSE 1,Ø,-159+16*n: DRAW  INVERS
E 1,-8,-16: DRAW 16*(7-n)+15,Ø:
DRAW  INVERSE 1;-8,16: DRAW  INV
ERSE 1;Ø,159-16*n
 82Ø PLOT 8*(n+1),16*(n+1): DRAW
  INVERSE 1;16*(6-n)+15,Ø
 83Ø PLOT 48,48: DRAW 31,Ø: PLOT
 48,32: DRAW 31,Ø
 84Ø GO TO 52Ø
 85Ø IF b=1 OR b=2 THEN  LET p=x
 86Ø IF b=3 OR b=4 THEN  LET p=y
 87Ø LET a=b: LET q=-q: RETURN
1ØØ1 IF q1=1 THEN  IF a$(a1,y1)=
"5" THEN  LET y1=y1+j: RETURN
1Ø1Ø IF q1=-1 THEN  IF a$(a1,x1)
="5" THEN  LET x1=x1+1: RETURN
1Ø2Ø IF x<x1 THEN  LET l=-1
1Ø3Ø IF x>x1 THEN  LET l=1
1Ø4Ø IF y<y1 THEN  LET j=-1
1Ø5Ø IF y>y1 THEN  LET j=1
1Ø6Ø IF q1=-1 THEN  GO TO 112Ø
1Ø7Ø LET b1= VAL a$(a1,y1)
1Ø8Ø IF x=x1 THEN  IF a$(a1,y1+j
) <> "Ø" THEN  LET y1=y1+j: RETU
RN
```

```
1Ø9Ø IF a$(b1,x1+1) <> "Ø" THEN
 LET a1=b1: LET q1=-1: LET x1=x1
+1: RETURN
11Ø1 IF a$(a1,y1+j) <> "Ø" THEN
 LET y1=y1+j: RETURN
111Ø LET j=-j: LET l=-1: GO TO 1
Ø9Ø
112Ø LET b1= VAL a$(a1,x1)
113Ø IF y=y1 THEN  IF a$(a1,x1+1
) <> "Ø" THEN  LET x1=x1+1: RETU
RN
114Ø IF a$(b1,y1+j) <> "Ø" THEN
 LET a1=b1: LET q1=1: LET y1=y1+
j: RETURN
116Ø IF a$(a1,x1+1) <> "Ø" THEN
 LET x1=x1+1: RETURN
117Ø LET j=-j: LET l=-1: GO TO 1
14Ø
2Ø15 INK 3
2Ø2Ø GO TO 2Ø3Ø+1Ø*d
2Ø4Ø FOR e=Ø TO 8: PRINT  AT 5+e
,4; INVERSE 1;m$(7*e+1 TO 7*e+7)
: NEXT e: RETURN
2Ø5Ø FOR e=Ø TO 8: PRINT  AT 5+e
,5; INVERSE 1;n$(1+e*6 TO 6+e*6)
: NEXT e: RETURN
2Ø6Ø FOR e=Ø TO 8: PRINT  AT 5+e
,5; INVERSE 1;o$(1+e*5 TO 5+e*5)
: NEXT e: RETURN
2Ø7Ø FOR e=Ø TO 7: PRINT  AT 5+e
,6; INVERSE 1;p$(1+e*4 TO 4+e*4)
: NEXT e: RETURN
2Ø8Ø FOR e=Ø TO 5: PRINT  AT 5+e
,7; INVERSE 1;q$(1+e*3 TO 3+e*3)
: NEXT e: RETURN
2Ø9Ø FOR e=Ø TO 3: PRINT  AT 5+e
,7; INVERSE 1;q$(19+e*2 TO 2Ø+e*
2): NEXT e: RETURN
21Ø1 PRINT  AT 5,7;"A"; AT 6,7;"
A"
211Ø RETURN
24Ø1 PAPER Ø: INK 4: PRINT  AT Ø
,Ø;e$: PRINT g$;g$;f$;h$;h$;h$;e
$;i$;i$;f$;g$;g$;g$;e$;i$;f$;h$;
h$;e$;g$;f$: RETURN
3Ø1Ø IF t1=1 AND t2=1 AND t3=1 T
HEN  IF  SCREEN$ (8,7)="M" THEN
 PRINT  AT 21,Ø;  FLASH 1;" You
have killed the minotaur ": GO T
O 3Ø3Ø
3Ø2Ø INK 5: PAPER 1: PRINT  AT 2
Ø,Ø; FLASH 1;" You have failed a
nd payed the    price of failure
"
3Ø3Ø FOR n=1 TO 2Ø: BEEP .2, INT
 ( RND *3Ø): NEXT n: INK 3: GO S
UB 2Ø4Ø: INK 6: PAPER Ø: STOP
321Ø DIM w(2,4)
322Ø FOR n=1 TO 4
323Ø LET w1= INT ( RND *22): LET
 w2= INT ( RND *32)
324Ø IF  SCREEN$ (w1,w2)="-" THE
N  LET w(1,n)=w1: LET w(2,n)=w2:
 NEXT n: GO TO 33ØØ
325Ø LET w1=w1+1: LET w2=w2+1: I
F w1>21 THEN  LET w1=Ø
326Ø IF w2>31 THEN  LET w2=Ø
327Ø GO TO 324Ø
33Ø1 LET y=w(1,4)+2: LET x=w(2,4
)+2
333Ø IF a$(3,w1+2)="2" OR a$(3,w
1+2)="1" THEN  LET b= VAL a$(3,w
1+2): LET q=-1: GO SUB 85Ø: GO T
O 17Ø
335Ø LET b= VAL a$(1,w2+2): LET
q=1: GO SUB 85Ø: GO TO 17Ø
341Ø IF y-2=w(1,1) AND x-2=w(2,1
) THEN  LET t1=1
342Ø IF y-2=w(1,2) AND x-2=w(2,2
) THEN  LET t2=1
343Ø IF y-2=w(1,3) AND x-2=w(2,3
) THEN  LET t3=1
344Ø GO TO 33Ø
4Ø1Ø LET m$="█BD CA██B    A██B"+
CHR$ 129+" "+ CHR$ 13Ø+"A█"+ CHR
$ 131+"F M G"+ CHR$ 131+"
      D   C  D    C  D    C "
4Ø2Ø LET n$="██DC"+ CHR$ 133+"██
▌"+ CHR$ 136+ CHR$ 132+ CHR$ 133
+"███"+ CHR$ 132+ CHR$ 136+" █
          D  C  D  C  D  C █DCD
C█"
4Ø3Ø LET o$="██  "+ CHR$ 133+"███
 "+ CHR$ 133+"█"+ CHR$ 131+ CHR$
 13Ø+"~"+ CHR$ 129+ CHR$ 131+"
   C   DC   D▄D C▄█DEC██DEC█"
4Ø4Ø LET p$="█DC██DC█    C  DC
DB  ABCDABCDA"
4Ø5Ø LET q$="B AF G   E E▌E"+ CH
R$ 133+"▌E"+ CHR$ 133+"▌"+ CHR$
133+"  "+ CHR$ 136+ CHR$ 132+"▌"
+ CHR$ 133
4Ø6Ø LET r$="█  █"+ CHR$ 131+"
"+ CHR$ 131+"=\/==  ="+ CHR$ 136
+"  "+ CHR$ 136+"▌"+ CHR$ 133+"▌
"+ CHR$ 133+"▌"+ CHR$ 133+"▌"+ C
HR$ 133
451Ø FOR n=Ø TO 55: READ a: POKE
 USR  CHR$ 144+n,a: NEXT n: RET
URN
452Ø DATA 127,127,127,127,127,12
7,127,127,254,254,254,254,254,25
4,254,254
453Ø DATA 1,1,1,1,1,1,1,1,128,12
8,128,128,128,128,128,128
454Ø DATA 24,24,24,24,24,24,24,2
4
455Ø DATA 254,254,254,254,Ø,Ø,Ø,
Ø,127,127,127,127,Ø,Ø,Ø,Ø
```

The Perfect Cover

Give your cassettes the professional look with this program from F A Marshall in Surrey.

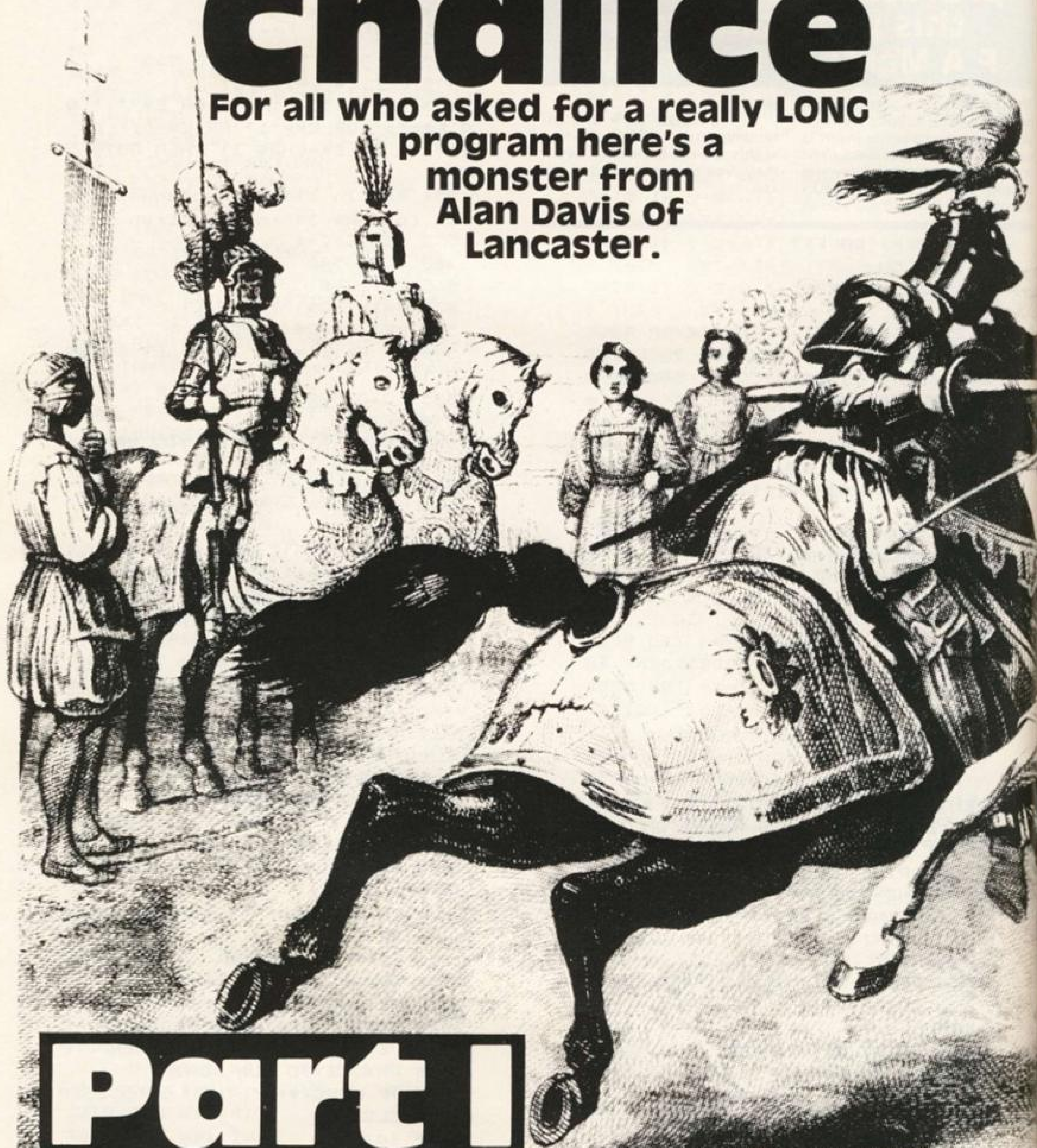
The coincidence that the ZX printer paper is exactly the right size to fit as a cassette insert is used to good effect here. The only problem seems to be where to stop! When entered and RUN the program will prompt you at all stages.

Lines 500-599 are interesting in that they ensure that words are not broken in two if they won't fit on the end of the line.

```
  10 REM "COVER"
  12 SLOW
  15 GOSUB 1000
  20 CLS
  25 PRINT AT 1,4;"WHATEVER YOU
WISH TO PRINT ","ON THE BACK(IND
EX)PART OF THE"," COVER SHOULD N
OW BE ENTERED"
  30 PRINT AT 10,4;"note TO CONT
 TO NEXT PHASE"
  32 PRINT AT 12,19;"ENTER QAZ"
  35 PRINT AT 15,11;"TO QUIT PRO
GRAMME"
  40 PRINT AT 17,19;"ENTER QQQ"
  45 GOSUB 2100
  50 CLS
 100 LET W$="-------------------
-------------"
 110 PRINT AT 0,0;W$
 120 GOSUB 500
 150 COPY
 200 CLS
 210 PRINT AT 0,4;"WHATEVER YOU
WISH TO PRINT","ON THE BACK(INDE
X)PART OF THE","COVER CAN NOW BE
 ENTERED","UPON ENTERING QAZ YOU
 WILL BE","ABLE TO PRINT ON THE
FRONT AREA","OF THE COVER"
 215 PRINT AT 10,4;"note    TO CO
NT TO NEXT PHASE"
 216 PRINT AT 12,19;"ENTER QAZ"
 217 PRINT AT 15,11;" TO QUIT PR
OGRAMME "
 218 PRINT AT 17,19;"ENTER QQQ"
 220 GOSUB 2100
 240 CLS
 275 PRINT AT 0,0;W$
 276 GOSUB 500
 280 PRINT AT 4,0;W$
 285 GOSUB 500
 290 PRINT AT 13,0;W$
 300 COPY
 499 GOTO 2500
 500 REM *****EDITOR**********
 515 INPUT Q$
 517 IF Q$="QAZ" THEN GOTO 590
 518 IF Q$="QQQ" THEN GOTO 2500
 520 IF LEN Q$>32 THEN GOTO 535
 525 PRINT Q$
 530 GOTO 515
 535 LET C=0
 540 IF C=32 THEN GOTO 565
 545 LET E$=Q$(33-C)
 550 IF Q$(33-C)=" " OR E$="," O
R E$="." OR E$=":" OR E$=";" OR
E$="?" OR E$=CHR$ 11 THEN GOTO 5
70
 551 REM In line 550 change CHR$
 11 to CHR$ 34 for Spectrum
 555 LET C=C+1
 560 GOTO 540
 565 LET C=0
 570 PRINT Q$( TO 32-C)
 575 LET Q$=Q$(33-C TO )
 580 IF Q$(1)=" " THEN LET Q$=Q$
(2 TO )
 585 GOTO 520
 599 RETURN
1000 PRINT AT 8,0;"DO YOU WANT I
NSTRUCTIONS (Y/N)"
1001 INPUT R$
1002 IF R$<>"Y" THEN RETURN
1010 CLS
1011 PRINT AT 1,10;"instructions
"
1013 PRINT AT 4,3;"THE PROGRAMME
 IS SELF";AT 6,0;"EXPLANATORY AS
 IT RUNS."
1015 PRINT AT 8,3;"TEXT EDITING
IS AUTOMATIC AS";AT 10,0;"IS COP
YING. THE FIRST AND LAST"
1017 PRINT AT 12,0;"DASH LINES A
RE CUT LINES, THE";AT 14,0;"OTHE
RS ARE FOLD LINES."
1019 PRINT AT 16,3;"DURING THE P
ROGRAMME IF YOU";AT 18,0;"WISH T
O QUIT THEN ENTER QQQ"
1020 PRINT AT 21,6;"PRESS ANY KE
Y TO RUN"
1030 IF INKEY$="" THEN GOTO 1030
1040 RETURN
2100 FOR T=1 TO 200
2150 NEXT T
2200 RETURN
2500 PRINT "THAT HAS PRODUCED YO
UR COVER. DOYOU WANT ANOTHER ? P
RESS ANY KEYTO RUN AGAIN."
2510 IF INKEY$="" THEN GOTO 2510
2520 CLS
2530 GOTO 5
```

The Golden Chalice

For all who asked for a really LONG program here's a monster from Alan Davis of Lancaster.



Part I

Well there's not much to say about this progrm except that I've played it and found it as challenging and absorbing as many of the adventure games being sold today.

To give a detailed breakdown would mean reducing the surprises you will find when playing it, so I'll just mention a few areas where problems may occur and recommend you enter it as you would play it. . . . . VERY carefully!

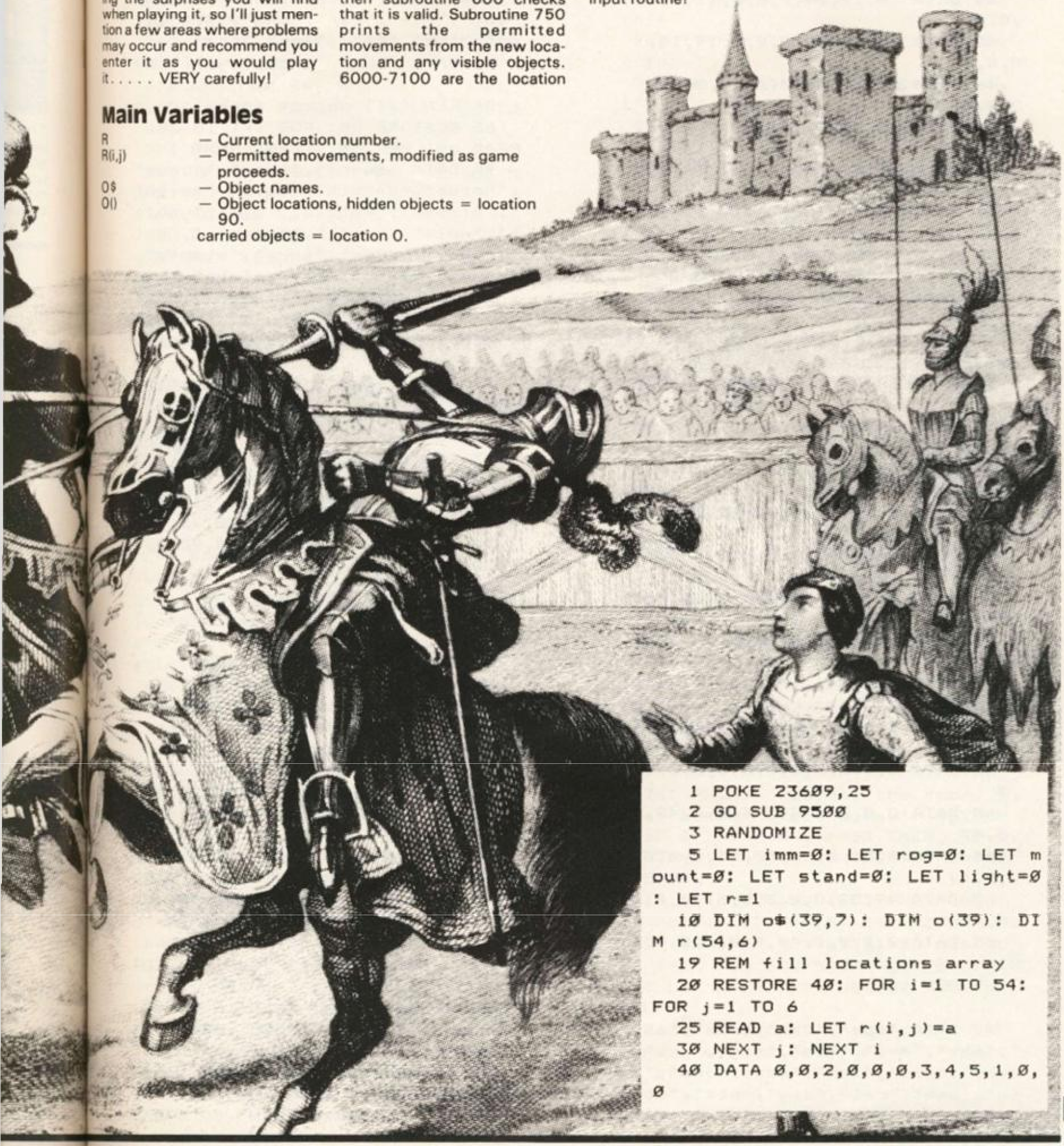
The main routine (lines 200-350) checks the input first for movement (single letter ?), then for the few verbs that single word entry is allowed and finally for the standard two word (Verb-Noun) entry.

If movement is required then subroutine 600 checks that it is valid. Subroutine 750 prints the permitted movements from the new location and any visible objects. 6000-7100 are the location subroutine lines.

When an action is required then each verb is allocated it's own subroutine which checks all the conditions necessary for that action to take place — provided that both verb and noun have been recognised by the input routine!

Main Variables

R — Current location number.
R(i,j) — Permitted movements, modified as game proceeds.
O$ — Object names.
O() — Object locations, hidden objects = location 90.
carried objects = location O.



```
  1 POKE 23609,25
  2 GO SUB 95ØØ
  3 RANDOMIZE
  5 LET imm=Ø: LET rog=Ø: LET m
ount=Ø: LET stand=Ø: LET light=Ø
: LET r=1
 1Ø DIM o$(39,7): DIM o(39): DI
M r(54,6)
 19 REM fill locations array
 2Ø RESTORE 4Ø: FOR i=1 TO 54:
FOR j=1 TO 6
 25 READ a: LET r(i,j)=a
 3Ø NEXT j: NEXT i
 4Ø DATA Ø,Ø,2,Ø,Ø,Ø,3,4,5,1,Ø,
Ø
```

```
  41 DATA Ø,2,Ø,Ø,Ø,Ø,2,Ø,Ø,Ø,Ø,
Ø
  42 DATA 6,Ø,8,2,Ø,Ø,7,Ø,8,5,Ø,
Ø
  43 DATA Ø,6,Ø,Ø,Ø,Ø,6,Ø,9,5,Ø,
Ø
  44 DATA 1Ø,17,14,8,Ø,Ø,11,9,12
,Ø,Ø,Ø
  45 DATA Ø,1Ø,Ø,Ø,Ø,Ø,Ø,14,16,1
Ø,Ø,Ø
  46 DATA Ø,12,Ø,Ø,Ø,Ø,12,2Ø,15,
9,Ø,Ø
  47 DATA 16,21,Ø,14,Ø,Ø,Ø,15,Ø,
Ø,Ø,Ø
  48 DATA 9,18,Ø,Ø,Ø,Ø,17,19,Ø,Ø
,Ø,Ø
  49 DATA 18,Ø,Ø,Ø,Ø,Ø,14,22,21,
Ø,Ø,Ø
  5Ø DATA 15,Ø,Ø,2Ø,Ø,Ø,2Ø,23,24
,Ø,Ø,Ø
  51 DATA 22,Ø,24,Ø,Ø,Ø,22,25,Ø,
23,Ø,Ø
  52 DATA 24,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
,Ø
  53 DATA 26,Ø,29,Ø,Ø,Ø,Ø,26,Ø,Ø
,Ø,Ø
  54 DATA Ø,27,Ø,26,Ø,3Ø,35,Ø,31
,Ø,29,39
  55 DATA 34,Ø,32,3Ø,Ø,Ø,Ø,Ø,33,
31,Ø,Ø
  56 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,
Ø
  57 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,37,Ø,3
1,Ø
  58 DATA Ø,38,Ø,36,Ø,Ø,37,Ø,Ø,Ø
,Ø,Ø
  59 DATA Ø,Ø,Ø,Ø,3Ø,Ø,Ø,42,39,Ø
,Ø,Ø
  6Ø DATA 39,Ø,44,42,Ø,Ø,4Ø,Ø,41
,Ø,Ø,Ø
  61 DATA Ø,44,Ø,39,Ø,Ø,43,Ø,Ø,4
1,Ø,Ø
  62 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,47
,Ø
  63 DATA Ø,Ø,Ø,Ø,48,46,Ø,Ø,49,Ø
,Ø,47
  64 DATA Ø,51,Ø,48,5Ø,53,Ø,Ø,Ø,
Ø,Ø,49
  65 DATA 49,52,Ø,Ø,Ø,Ø,51,Ø,Ø,Ø
,Ø,Ø
  66 DATA Ø,Ø,Ø,Ø,49,Ø,Ø,Ø,Ø,53,
Ø,Ø
  69 REM verbs

  7Ø DATA "ope","tak","exa","ent
","eat","dri","lea","fig","giv",
"mou","dis","loo","row","drp","l
ig","bre","cal","dig","sta","cli
","buy"
  71 DATA "say"
  79 REM nouns
  8Ø DATA "swo","foo","pur","hor
","her","cav","kni","str","wat",
"lam","gob","wol","tro","rog","h
ut","lad","doo","sho","sta","che
","oar","key","lak","win","boo",
"eag","ste","cur","cha","liq","m
on","tem","ben","led","cra"
  81 DATA "boa","map","ogr","roo
"
  84 REM fill object array
  85 RESTORE 9Ø: FOR i=1 TO 39:
READ a$: LET o$(i)=a$: NEXT i
  9Ø DATA "sword","food","purse"
,"horse","hermit","cave","knight
","stream","water","lamp","goble
t","wolf","troll","roggoth","hut
","lady","door","shop","stable",
"chest","oars","key","lake","win
dow","book","eagle","steps","cur
tain","chalice","liquid","money"
,"temple","bench","ledge","crack
"
  91 DATA "boat","map","ogre","r
oom"
  94 REM set up object locations
  95 RESTORE 1ØØ: FOR i=1 TO 39:
 READ a: LET o(i)=a: NEXT i
 1ØØ DATA 11,4,9Ø,3,7,9Ø,11,9Ø,9
Ø,9Ø,25,16,25,21,12,9Ø,9Ø,9Ø,9Ø,
1,9Ø,9Ø,9Ø,5Ø,52,9Ø,9Ø,54,9Ø,9Ø,
9Ø,9Ø,44,9Ø,9Ø,38
 1Ø1 DATA 19,39,9Ø
 15Ø BORDER 7: PAPER 7: INK Ø: C
LS
 154 REM print instructions
 155 GO SUB 96ØØ
 197 REM describe init. location
 198 GO SUB 6ØØØ+2Ø*r
 199 REM input routine
 2ØØ LET move=Ø: LET ill=Ø: INPU
T "WHAT DO YOU WISH TO DO ?"';z$
 2Ø5 IF z$="i" THEN  GO SUB 55ØØ
: GO TO 2ØØ
 21Ø IF LEN z$<3 THEN  GO SUB 6Ø
Ø
 211 IF r<26 THEN  GO SUB 9ØØ
 212 IF r=21 AND NOT rog THEN  L
ET rog=1
 22Ø IF move OR ill THEN  GO TO
2ØØ
 221 IF z$="stop" THEN  RUN
 222 IF z$="wait" THEN  GO SUB 4
1ØØ: GO TO 2ØØ
 223 IF z$="dismount" THEN  GO S
UB 2ØØØ: GO TO 2ØØ
 23Ø IF z$="look" THEN  GO SUB 5
```

```
900: GO SUB 6000+20*r: GO TO 200
 240 IF z$( TO 3)="   " THEN  GO
 TO 200
 250 GO SUB 500
 260 RESTORE 70: FOR v=1 TO 22:
READ v$: IF v$=w$(1) THEN  GO TO
 270
 261 NEXT v
 265 IF v$<>w$(1) THEN  GO SUB 5
000: GO TO 200
 270 RESTORE 80: FOR n=1 TO 39:
READ n$: IF n$=w$(2) THEN  GO TO
 300
 271 NEXT n
 275 IF n$<>w$(2) THEN  GO SUB 5
000: GO TO 200
 300 GO SUB 900+100*v
 350 GO TO 200
 499 REM decode input
 500 DIM w$(2,3)
 510 IF z$="" THEN  RETURN
 520 FOR z=1 TO LEN z$
 530 IF z$(z)=" " THEN  NEXT z:
RETURN
 540 LET w$(1)=z$(z TO )
 550 FOR z=LEN z$ TO z+2 STEP -1
 560 IF z$(z)=" " THEN  NEXT z:
RETURN
 570 IF z$(z-1)<>" " THEN  NEXT
z: RETURN
 580 LET w$(2)=z$(z TO ): RETURN

 599 REM movement
 600 IF ((r=2 AND (z$="w" OR z$=
"n")) OR (r=6 AND z$="n") OR (r=
12 AND z$="n") OR (r=21 AND z$="
e") OR (r=24 AND z$="s")) AND mo
unt THEN  GO TO 670
 605 IF z$="n" AND r(r,1) THEN
LET move=1: LET r=r(r,1): PRINT
'"You go north."
 610 IF z$="s" AND r(r,2) THEN
LET move=1: LET r=r(r,2): PRINT
'"You go south."
 620 IF z$="e" AND r(r,3) THEN
LET move=1: LET r=r(r,3): PRINT
'"You go east."
 630 IF z$="w" AND r(r,4) THEN
LET move=1: LET r=r(r,4): PRINT
'"You go west."
 640 IF (z$="up" OR z$="u") AND
r(r,5) THEN  LET move=1: LET r=r
(r,5): PRINT '"You go up."
 650 IF z$="d" AND r(r,6) THEN
LET move=1: LET r=r(r,6): PRINT
'"You go down."
 660 IF move THEN  GO SUB 5900:
GO SUB 6000+20*r: GO SUB 4000: R
ETURN
 670 PRINT '"You can't do that."
: LET ill=1: RETURN
 749 REM print directions & objs
 750 PRINT '"You may go:"
 755 IF r(r,1) THEN  PRINT " nor
th";
 760 IF r(r,2) THEN  PRINT " sou
th";
 765 IF r(r,3) THEN  PRINT " eas
t";
 770 IF r(r,4) THEN  PRINT " wes
t";
 775 IF r(r,5) THEN  PRINT " up"
;
 780 IF r(r,6) THEN  PRINT " dow
n";
 785 IF NOT r(r,1) AND NOT r(r,2
) AND NOT r(r,3) AND NOT r(r,4)
AND NOT r(r,5) AND NOT r(r,6) TH
EN  PRINT  "nowhere"
 790 LET obj=0: PRINT ''"You see
:": FOR i=1 TO 39: IF o(i)=r THE
N  PRINT " a ";o$(i): LET obj=1
 795 NEXT i: IF NOT obj THEN  PR
INT " nothing special."
 796 RETURN
 899 REM roggoth encounter
 900 IF NOT imm AND rog AND move
 AND r<>21 AND r<26 THEN  PRINT
'"The roggoth enters.": LET o(14
)=r: LET r(21,3)=26
 905 IF NOT imm AND rog AND r<>2
1 AND (NOT mount OR NOT move) TH
EN  PRINT '"The roggoth kills yo
u.": GO TO 9990
 920 RETURN
 998 REM action conditions
 999 REM open
1000 IF n=17 AND r=12 AND o(10)=
90 THEN  GO SUB 5010: LET r(r,1)
=13: RETURN
1005 IF n=17 AND r=31 AND NOT o(
22) THEN  GO SUB 5010: LET r(r,6
)=36: PRINT '"Beyond the door, s
tone steps"'"lead down.": RETURN
1010 IF n=17 AND r=53 THEN  GO S
UB 5600: IF a THEN  GO SUB 5005:
 RETURN
1015 IF n=17 AND r=53 THEN  GO S
UB 5010: LET r(r,3)=54: RETURN
1020 IF n=28 AND o(28)=54 AND r=
54 THEN  GO SUB 5010: LET o(29)=
54: LET o(28)=90: GO SUB 7082: R
ETURN
1025 IF n=20 AND r=1 AND o(3)=90
 THEN  GO SUB 5010: PRINT '"In t
he chest is a purse"'"containing
 coins.": LET o(3)=1: RETURN
1026 IF n=20 AND r=1 AND o(3)<>1
```

```
 THEN  GO SUB 5Ø1Ø: RETURN
1Ø3Ø IF n=25 AND (r=52 OR NOT o(
25)) THEN  GO SUB 5Ø1Ø: RETURN
1Ø35 GO SUB 5ØØ5: RETURN
1Ø99 REM take
11ØØ IF n=1 AND o(1)=11 AND o(7)
=11 THEN  GO SUB 5Ø15: PRINT '"T
he wounded knight cries:"'"A thi
ef shall not have my sword!"''"H
e reaches up,and with his"'"last
 remaining strength plunges"'"a
dagger through your heart."'" Yo
u die instantly.": GO TO 999Ø
11Ø5 IF n=1 AND o(1)=r THEN  GO
SUB 5Ø15: LET o(1)=Ø: RETURN
1115 IF n=2 AND o(2)=r AND r=4 T
HEN  GO SUB 5Ø15: PRINT '"You ar
e arrested for stealing"'"the fo
od. You spend the rest of"'"your
 life in the king's dungeon.": G
O TO 999Ø
1116 IF n=2 AND o(2)=r THEN  GO
SUB 5Ø15: LET o(2)=Ø: RETURN
112Ø IF n=3 AND o(3)=r THEN  GO
SUB 5Ø15: LET o(3)=Ø: RETURN
113Ø IF n=4 AND r=3 AND o(4)=3 T
HEN  GO SUB 5Ø15: PRINT '"You ar
e arrested as a horse"'"thief. Y
ou spend the rest of"'"your life
 in the king's dungeon.": GO TO
999Ø
```



```
1131 IF n=4 AND o(4)=r THEN  GO
SUB 5Ø15: LET o(4)=Ø: RETURN
1135 IF n=9 AND NOT o(11) AND (r
=11 OR r=1Ø OR r=17 OR r=9 OR r=
18 OR r=19 OR r=24) THEN  GO SUB
 5Ø15: LET o(9)=Ø: RETURN
114Ø IF n=1Ø AND r=o(1Ø) THEN  G
O SUB 5Ø15: LET o(1Ø)=Ø: RETURN
1145 IF n=11 AND r=o(11) AND o(1
3)=9Ø THEN  GO SUB 5Ø15: LET o(1
1)=Ø: RETURN
115Ø IF n=21 AND r=o(21) THEN  G
O SUB 5Ø15: LET o(21)=Ø: RETURN
1155 IF n=22 AND r=o(22) THEN  G
O SUB 5Ø15: LET o(22)=Ø: RETURN
116Ø IF n=25 AND r=o(25) THEN  G
O SUB 5Ø15: LET o(25)=Ø: RETURN
1165 IF n=29 AND r=o(29) THEN  G
O SUB 5Ø15: LET r(54,4)=Ø: LET o
(29)=Ø: RETURN
117Ø IF n=33 AND r=o(33) THEN  G
O SUB 5Ø15: LET o(33)=Ø: RETURN
1175 IF n=36 AND r=o(36) THEN  G
O SUB 5Ø15: LET o(36)=Ø: LET r(3
7,3)=45: RETURN
118Ø IF n=37 AND r=o(37) THEN  G
O SUB 5Ø15: LET o(37)=Ø: RETURN
119Ø GO SUB 5ØØ5: RETURN
1199 REM examine
12ØØ IF n=17 AND r=53 THEN  GO S
UB 5Ø2Ø: PRINT '"Time has worn a
way much of the"'"carved work on
 the door, but you"'"are able to
 make out a few"'"incomplete wor
ds:"''"  ET HE W O IS WIT OUT"'"
COM  WI HIN.": RETURN
12Ø5 IF n=24 AND r=5Ø THEN  GO S
UB 5Ø2Ø: PRINT '"Looking through
 the window, you"'"see below the
 vast expanse of"'"the undergrou
nd lake. Above is"'"the great va
ulted cavern roof,"'"permeating
all with its pale"'"greenish lig
ht.": RETURN
121Ø IF n=25 AND (r=o(25) OR NOT
 o(25)) THEN  GO SUB 5Ø2Ø: PRINT
 '"You read the words:"'"HE WHO
CALLS THE EAGLE OVER"'"WATER SHA
LL RETURN.": RETURN
1215 IF n=29 AND (r=o(29) OR NOT
 o(29)) THEN  GO SUB 5Ø2Ø: PRINT
 '"The chalice is full, almost t
o"'"the brim, of a clear golden"
'"liquid with heady bouquet.": R
ETURN
122Ø IF n=34 AND o(22) AND r=42
AND (o(33)=42 OR NOT o(33)) AND
```

```
stand THEN  GO SUB 5020: PRINT '
"On the ledge is an old, rusty"'
"key.": LET o(22)=42: RETURN
1225 IF n=35 AND r=38 THEN  GO S
UB 5020: IF o(21)=90 THEN  PRINT
 '"Wedged into the crack is a pa
ir"'"of oars.": LET o(21)=38: RE
TURN
1230 IF n=37 AND NOT o(37) THEN
 GO SUB 5020: PAUSE 100: GO SUB
9100: RETURN
1290 PRINT "There's nothing spec
ial to see.": RETURN
1299 REM enter
1300 IF mount THEN  GO SUB 5005:
 RETURN
1302 IF r=2 THEN  PRINT '"Which
way?": RETURN
1305 IF r=6 THEN  LET r=r(r,1):
GO SUB 5025: RETURN
1310 IF r=12 AND r(r,1) THEN  LE
T r=r(r,1): GO SUB 5025: RETURN
1315 IF r=24 THEN  LET r=r(r,2):
 GO SUB 5025: RETURN
1320 IF r=21 AND r(r,3) THEN  LE
T r=r(r,3): GO SUB 5025: RETURN
1325 IF r=31 AND r(r,6) THEN  LE
T r=r(r,6): GO SUB 5025: RETURN
1330 IF r=37 THEN  PRINT '"Which
 way?": RETURN
1335 IF r=53 AND r(r,3) THEN  LE
T r=r(r,3): GO SUB 5025: RETURN
1390 GO SUB 5005: RETURN
1399 REM eat
1400 IF n=2 AND NOT o(2) THEN  P
RINT ' "You eat the food.": LET
o(2)=90: LET r(26,1)=28: LET r(2
6,2)=27: LET r(26,3)=29: LET r(2
6,4)=21: RETURN
1490 GO SUB 5005: RETURN
1499 REM drink
1500 IF n=9 AND (r=11 OR r=10 OR
 r=9 OR r=17 OR r=18 OR r=19 OR
r=24 OR r=37 OR r=45 OR r=46 OR
NOT o(9)) THEN  GO SUB 5030: LET
 o(9)=90: RETURN
1505 IF n=30 AND NOT o(29) THEN
 GO SUB 5030: PRINT '" For a mom
ent nothing seems to"'"happen -
but then a golden"'"warmth begin
s to spread through"'"your body
as the ancient"'"enchantment tak
es effect."'" You are immortal..
.": LET imm=1: RETURN
1515 IF n=8 OR n=23 THEN  PRINT
'"Don't be greedy!": RETURN
1590 GO SUB 5005: RETURN
1599 REM leave
1600 IF r=1 OR r=3 THEN  LET r=2
: GO SUB 5035: RETURN
1605 IF r=7 OR r=13 THEN  LET r=
r(r,2): GO SUB 5035: RETURN
1610 IF r=25 THEN  LET r=24: GO
SUB 5035: RETURN
1615 IF r=26 THEN  LET r=21: GO
SUB 5035: RETURN
1620 IF r=38 THEN  LET r=37: GO
SUB 5035: RETURN
1690 GO SUB 5005: RETURN
1699 REM fight
1700 IF (n=5 AND r=7) OR (n=7 AN
D r=11) THEN  GO SUB 5040: GO SU
B 5045: RETURN
1705 IF n=12 AND o(12)=16 THEN
GO SUB 5040: PRINT'"The wolf is
mortally injured"'"and runs off
howling.": LET o(12)=90: RETURN
1710 IF n=13 AND o(13)=25 THEN
GO SUB 5040: PRINT '"You hack of
f the troll's head.": LET o(13)=
90: RETURN
1715 IF n=14 AND o(14)=r THEN  P
RINT '"I hope you're sure about
this!": PAUSE 50: GO SUB 5040: P
RINT '"With one blow the roggoth
 kills"'"you.": GO TO 9990
1720 IF n=38 AND r=o(38) THEN  G
O SUB 5040: GO SUB 5700: RETURN
1790 GO SUB 5005: RETURN
1799 REM give
1800 IF n=9 AND r=11 AND o(7)=11
 AND NOT o(9) THEN  LET o(7)=90:
 LET o(9)=90: LET o(11)=11: GO S
UB 5050: PRINT '"The knight drin
ks the water"'"gratefully,and sa
ys:"'"  Thou art a good knight a
nd"'"true. I am not long for thi
s"'"world. Take my sword, and us
e it"'"to rid these lands of evi
l."'"With a sigh, the knight die
s."'"To your astonishment, four
men,"'"dressed in white, enter."
'"They take the body of the knig
ht"'"and leave.": LET o(7)=90: R
ETURN
1805 IF n=10 AND r=13 THEN  PRIN
T '"You ask for a ";o$(10)'"The
lady says: I give that"'"which I
 may give."'"She gives you a ";o
$(10): LET o(10)=0: LET r(12,1)=
0: RETURN
1810 IF n=2 AND r=7 AND o(5)=7 T
HEN  GO SUB 5050: LET o(2)=90: P
RINT '"The hermit accepts gratef
ully.": RETURN
1890 GO SUB 5005: RETURN
1899 REM mount
1900 IF (o(4)=r OR NOT o(4)) AND
```

```
r<>3 AND NOT mount AND n=4 THEN
 PRINT '"You mount the horse.":
 LET o(4)=Ø: LET mount=1: RETURN
199Ø GO SUB 5ØØ5: RETURN
1999 REM dismount
2ØØØ IF mount  THEN  PRINT '"You
 dismount": LET o(4)=r: LET moun
t=Ø: RETURN
2Ø9Ø GO SUB 5ØØ5: RETURN
2Ø99 REM look
219Ø GO SUB 5ØØ5: RETURN
2199 REM row
22ØØ IF r=45 AND NOT o(21) THEN
 PRINT '"You row the boat.": LET
 r=46: GO SUB 59ØØ: GO SUB 6ØØØ+
2Ø*r: RETURN
229Ø GO SUB 5ØØ5: RETURN
2299 REM drop
23ØØ IF NOT o(29) THEN  PRINT '"
You replace the ";o$(n): LET r(5
4,4)=53: LET o(29)=54: RETURN
23Ø5 IF NOT o(n) THEN  PRINT '"Y
ou drop the ";o$(n): LET o(n)=r:
 RETURN
239Ø GO SUB 5ØØ5: RETURN
2399 REM light
24ØØ IF NOT o(1Ø) AND n=1Ø THEN
 PRINT '"You light the ";o$(1Ø):
 LET light=1: RETURN
249Ø GO SUB 5ØØ5: RETURN
2499 REM break
259Ø GO SUB 5ØØ5: RETURN
2599 REM call
26ØØ IF imm AND (r=46 OR r=47 OR
 r=48) AND n=26 THEN  PRINT '"Yo
u call the ";o$(n): PAUSE 5Ø: PR
INT '"After some minutes you hea
r the"'"beating of gigantic wing
s."'"A huge eagle descends from
the"'"sky, catches you in its ta
lons"'"and soars upward again."'
"The eagle flies across the vast
"'"lake and deposits you careful
ly"'"on the western shore.": LET
 r=37: RETURN
269Ø GO SUB 5ØØ5: RETURN
2699 REM dig
279Ø GO SUB 5ØØ5: RETURN
2799 REM stand
28ØØ IF n=33 AND (NOT o(33) OR o
(33)=42) AND r=42 THEN  LET stan
d=1
281Ø IF n=33 AND (NOT o(33) OR o
(33)=r) THEN  PRINT '"You stand
on the ";o$(33);".": RETURN
289Ø GO SUB 5ØØ5: RETURN
2899 REM climb
29ØØ GO SUB 28ØØ: RETURN
299Ø GO SUB 5ØØ5: RETURN
2999 REM buy
3ØØØ IF (n=4 OR n=2) AND o(n)=r
AND NOT o(3) THEN  PRINT '"You b
uy the ";o$(n): LET o(n)=Ø:RETURN
3Ø9Ø GO SUB 5ØØ5: RETURN
3Ø99 REM say
31ØØ IF n=1Ø AND r=13 THEN  GO S
UB 18Ø5: RETURN
31Ø5 PRINT '"You mumble somethin
g.": RETURN
4ØØØ RETURN
4Ø99 REM wait
41ØØ IF r=24 THEN  PRINT '" The
troll emerges from the"'"cave. H
e looks about, but does"'"not se
e you standing among the"'"trees
. He goes north.": LET o(13)=9Ø:
 RETURN
419Ø PRINT '"Nothing happens.":
RETURN
4999 REM stock responses
5ØØØ PRINT '"I don't understand
that.": RETURN
5ØØ5 PRINT '"You can't do that."
: RETURN
5Ø1Ø PRINT '"You open the ";o$(n
): RETURN
5Ø15 PRINT '"You take the ";o$(n
): RETURN
5Ø2Ø PRINT '"You examine the ";o
$(n): RETURN
5Ø25 PRINT "You enter.": GO SUB
59ØØ: GO SUB 6ØØØ+2Ø*r: RETURN
5Ø3Ø PRINT '"You drink the ";o$(
n): RETURN
5Ø35 PRINT '"You leave.": GO SUB
 59ØØ: GO SUB 6ØØØ+2Ø*r: RETURN
5Ø4Ø PRINT '"You fight the ";o$(
n): RETURN
5Ø45 PRINT '"The ";o$(n);" makes
 no real"'"effort to resist,and
dies.": LET o(n)=9Ø: LET o(1)=9Ø
: RETURN
5Ø5Ø PRINT '"You give the ";o$(n
): RETURN
5Ø55 PRINT '"You wait.": RETURN
5499 REM inventory
55ØØ LET o=Ø: PRINT '"You have w
ith you:"
55Ø1 FOR i=1 TO 39
55Ø2 IF NOT o(i) THEN  PRINT "
  ";o$(i): LET o=1
55Ø4 NEXT i
55Ø5 IF NOT o THEN PRINT "nothing"
55Ø6 RETURN
```

Part Two Next Issue

Part two of The Golden Chalice will appear in the next mind-boggling issue of *ZX Computing*.

48K SPECTRUM

TASWORD TWO THE WORD PROCESSOR

64 CHARACTERS PER LINE ON THE SCREEN AND TO PRINTERS!
MICRODRIVE COMPATIBLE — instructions supplied

"The number of on-screen prompts, together with the excellent manual, make it ideal – even for an absolute beginner."

PERSONAL COMPUTER WORLD September 1983

"What makes a word processor more or less versatile is its control features. Tasword Two offers an impressive selection and the tutor program succeeds in demonstrating them admirably."

ELECTRONICS AND COMPUTING November 1983

"Tasword is showing a degree of sophistication that business computers took many years to develop."

WHAT MICRO? October 1983

TASWORD TWO The Word Processor*

Your Spectrum becomes a **professional** word processor with TASWORD TWO. TASWORD TWO gives you an amazing **64 characters per line** on your screen. This is ideal for standard A4 paper and TASWORD TWO prints your text just as it appears on your screen.

Tasword Two drives the following interfaces:

Cobra RS232 I/O Port	*ADS Interface*
Euroelectronics Interface	*Kempston Interface*
Hilderbay Interface	*Morex Interface*
Sinclair ZX Interface 1	*Tasman Interface*

The same program drives these interfaces. A short easy to follow set of instructions takes you through setting up your Tasword Two to drive the interface you have or choose to buy. Tasword Two also drives the ZX printer.

Tasword Two is readily adapted for the microdrives to give super-fast saving and loading of both program and text. The microdrive instructions are supplied with the Tasword Two manual.

£13·90 fully inclusive mail order price.

TASWORD TWO TUTOR

TASWORD TWO comes complete with a manual and a cassette. The cassette contains your TASWORD TWO **and** TASWORD TWO TUTOR. This teaches you word processing using TASWORD TWO. Whether you have serious applications or simply want to learn about word processing, TASWORD TWO and TASWORD TWO TUTOR make it easy and enjoyable.

TASWORD TWO £2 Demonstration Cassette

See for yourself the powerful features of TASWORD TWO. Send just £2 for the Tasword Two demonstration cassette. A voucher is included which gives you £1 off the price of TASWORD TWO.

All prices include VAT and post and packaging

* Available from larger branches of Boots

For further information on all these products send an s.a.e. with "Tasman Brochure" written on the flap.

Send cheque/P.O. or Access number with order.
Telephone Access orders: Leeds (0532) 438301

TASPRINT

A must for dot matrix printer owners! Print your program output and listings in a choice of five impressive print styles. Drives all the printer interfaces listed under Tasword Two and all dot matrix printers with bit image graphics capabilities. You can also use TASPRINT to print Tasword Two text files. TASPRINT gives your output originality and style! Send s.a.e. for brochure which includes TASPRINT output.

£9.90 fully inclusive mail order price

TASWIDE – 64 characters per line!

A machine code utility program, TASWIDE **doubles** the information that your own programs can display. Make a simple change to your print statements and your output appears on the screen at 64 characters per line instead of the normal 32. Both print sizes can be mixed on the screen. 16K and 48K versions supplied on the same cassette.

£5·50 fully inclusive mail order price

TASMATH 1

Mathematics education. Three programs on one cassette:
TASIMEQ – simultaneous equations
TASQUAD – quadratic equations
TAS-TRIG – trigonometry of right angled triangles.
Available for the 16K and 48K Spectrum and for the BBC model B.

£5.95 fully inclusive mail order price

TASMAN PARALLEL PRINTER INTERFACE

Plug into your Spectrum and drive any printer fitted with the Centronics standard parallel interface. Supplied complete with ribbon cable and driving software. The cassette includes LLIST, LPRINT, and **text** screen copy software for **all** centronics printers. The user changeable interface software makes it easy to send control codes to your printer using the method so successfully pioneered with Tasword Two. The cassette **also** contains fast machine code high resolution screen copy software for Epson, Star, Seikosha, Shinwa, and Tandy Colour Graphic (in colour!) printers. Send s.a.e. for brochure which includes sample print-outs and a full list of printers supported by screen software. Compatible with microdrives/ZX Interface 1.

£39.90 fully inclusive mail order price

TASMAN RS232 PRINTER INTERFACE

Specification and software as above but drives printers fitted with the RS232 standard interface. A low cost route to printing – especially suitable for use with the Tandy Colour Graphic printer and the Brother portable typewriter/printers. Supplied complete with cable – please specify whether 4 pin DIN or 25 way D plug required.

£38 fully inclusive mail order price

TASMAN SOFTWARE

Dept ZXC
SPRINGFIELD HOUSE
HYDE TERRACE, LEEDS LS2 9LN

Spectrum lessons

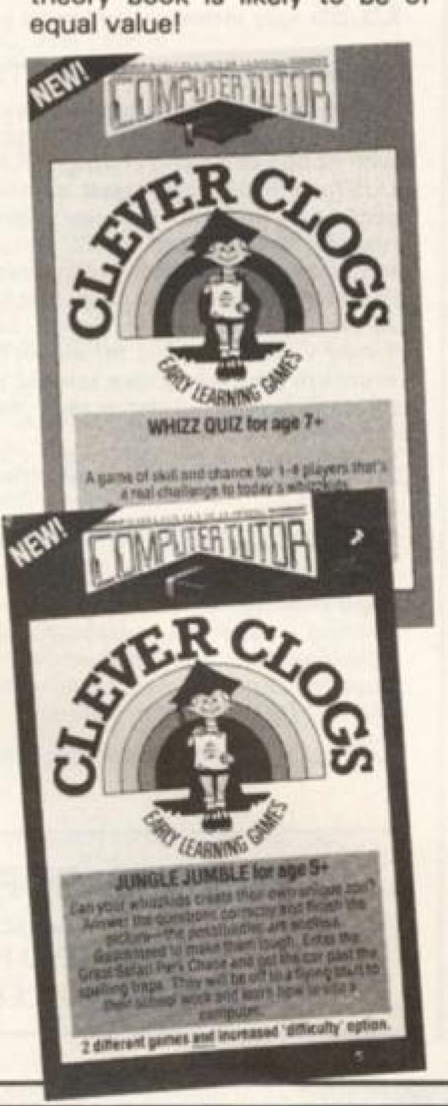
A mixed bag for Mike Edmunds to criticise and commend as he sees fit.

Once upon a time if you wanted a game to play then you visited your local software shop. If you required something in the educational line you had to resort to mail-order or search out a specialist supplier. Nowadays, however, we find an increasing number of 'educational' programs rubbing shoulders with our favourite arcade game. Whereas the use of the computer as a valuable educational tool within the school is undeniable, does the rapidly expanding range of 'home learning' software successfully bridge the gap between home and school? This month some of the currently available programs are examined and assessed. . .

Learning to Read Music

Firstly, from Rose Software, who have a reputation for quality educational software, comes 'Learning To Read Music'. This contains four programs to help develop the skills needed in order to read music. Technically, this is a commendable program, the layout and presentation being of a high standard. Colour and graphics are well used and the sound, given the limitations of the Spectrum's tiny speaker is adequate. The first program is a game for beginners called 'Treble Clef' which aids pitch discrimination and musical literacy. Notes must be recognised from the screen. Corrections are given and a 'help' facility is available, if required. This program is straightforward but, for some reason is followed by 'Scale Position', designed for the older student preparing for ABSM Grade V Theory. This is a much more complex program where the user is required to name a given key signature and then name notes, such as the Tonic or Dominant, in relation to that key.

'Note Values' teaches and tests the relative value of certain notes and rests but the procedure seems to have more of a mathematical rather than a musical emphasis. A rhythmic sequence is built up by the student and is then played, showing how successful the learning has been. 'Keyboard', the final program is designed for keyboard beginners and shows the relationship between notes on the music stave and notes on the keyboard. It also explains sharps and flats. All in all, this is a comprehensive package which aims to cater for a fairly wide age and ability range, but in my opinion a computer is no substitute for a piano and a good theory book is likely to be of equal value!



Whizz Quiz

The second title comes from Computertutor, who can boast, as part of their family based team, reviewers and co-designers aged from three to eight! The program is called 'Whizz Quiz' and is one of eight 'Clever Clogs' titles, this one aimed at children of 7+. Basically, this is a computerised version of Ludo with a selection of general knowledge questions appearing at random. An animated character, Clever Clogs, is the guide throughout the game.

The computer throws the dice for each turn, however, don't play just against Clever Clogs — he seems to get very high throws and extra turns much more frequently than I did! The screen gets pretty crowded when four play and the hectic race around the board is obviously part of the appeal for children. Personally, I didn't like the randomness of the questions, but my class of 7-9 year olds loved it. There is an option for parents/teachers to change the questions but this is a time-consuming task! Nicely presented and fun for the home, but likely to be of less value within a class-room.

Spacewreck

Also for the younger child is 'Spacewreck', from Premier Software. This has been written by Primary School Teachers, it's aim being to motivate creative writing. Expecting something of real educational value I approached this program eagerly, but I was sadly disappointed. This is a '...computer designed adventure with visual stimulus and words to help the child. Using these, the children are asked to write a story based upon their adventure.' Unfortunately this has been very poorly written and

SPACEWRECK



48K SPECTRUM

I can see little, if any, value in it as an educational program. Compared with current Spectrum programs the graphics here are very poor and very slow!... graphic blocks move jerkily across the screen and there has been no attempt to use UDG's (captial letters being used to represent people and monsters). There was also a serious bug in the review copy, the 'adventure' forming a repeating loop after about 10 minutes. Premier Software offer a marking and assessment service for £1.50, but any teacher who needs to rely on programs such as this in order to motivate children probably shouldn't be teaching at all! The whole program needs to be re-evaluated and redesigned.

Antony and Cleopatra

From Akadimias Educational Software comes a practical revision aid for students preparing for Secondary Level Examinations. This is a man-driven program which contains six units based upon Shakespeare's 'Antony and Cleopatra! Each unit tackles a different aspect of the set text e.g. Plot, Background,

SPECTRUM 48K

ΑΚΑΔΙΜΙΑΣ

FOUNDATION SERIES

ANTONY AND CLEOPATRA by William Shakespeare

program by
Dr. Margarette Smith

Department of English
University College of North Wales
Bangor

Imagery etc. and contains 120+ questions and answers. The questions have been devised by University lecturers and are very comprehensive. The program offers Revision, Sequential and Test Modes and each has 4 time/skill levels to introduce a games element. The aim is to reinforce both cognitive, analytical and discursive skills but is not intended as a substitute for reading or discussion. The structure is extremely thorough and the games element livens up a rather 'dry' format but, apart from that, little real use seems to have been made of the computer. Nevertheless, this will undoubtedly be of considerable value to students using this set text, although it must be noted that for the price you could probably buy several good revision books!

First Aid

....and now for something completely different! From the company that brought you 'How long have you got?' comes 'First Aid' (Draw your own conclusions!) Eastmead Computer Systems have produced a comprehensive reference package suitable for use in '... classrooom, home and surgery.' (As an educational tool this is well put together although I'd hate to see my doctor using it!) It is a computerised 'Home Doctor' that aims, through a series of questions, to give you guidance upon the course of action required for a vast range of symptoms. Twenty areas, in four programs, are covered, including Accidental Injury, Artificial Respiration, Bleeding, Electric Shock, Fever etc. Nice use of colour helps an all-text display but it is rather disconcerting, at times, to be told 'Emergency help needed'.... One cause for complaint, in an otherwise admirable package, is that from time to time, two or more questions are asked simultaneously which may require differing answers. Unfortunately only one response at a time is allowed.

In conclusion it seems that an increasing amount of software can be used successfully in more than one area, but your needs should be very clear before you buy!

'Learning to read music'
SPECTRUM 16K
Rose Software,
148 Widney Lane, Solihull, W. Midlands
£5.95

Whizz Quiz
SPECTRUM 48K
Computertutor,
1 Golden Square, London W1R 3AB
£6.50

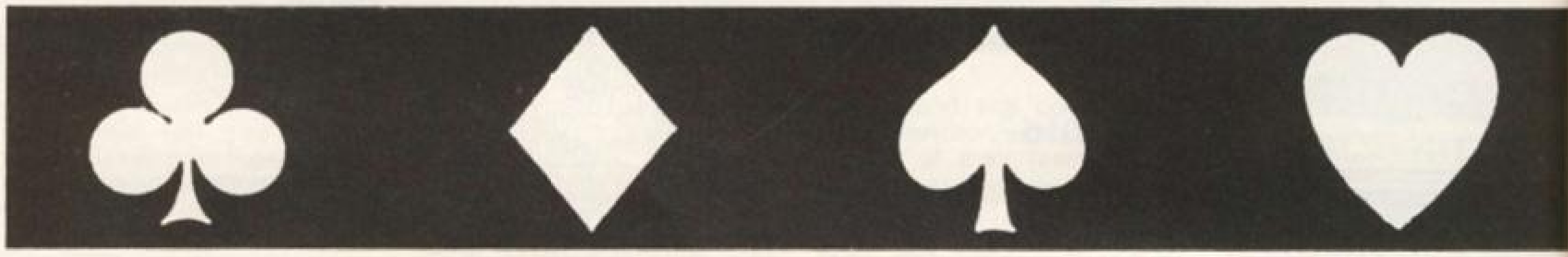
First Aid,
SPECTRUM 48K
Eastmead Computer Systems Ltd, Eastmead House, Lyon Way, Camberley, Surrey.
£4.60

Spacewreck
SPECTRUM 48K
Premier Software, 1 Holington Rd., Upper Team, Stoke-on-Trent.

'Antony and Cleopatra' Shakespeare
SPECTRUM 48K
Akadimias, Sussex Software, Townsend, Poulshot, Devizes, Wiltshire
£10.00

Card Corner — Patience II

Colin Gooch presents a variation known as 8 card patience in part 2 of his series.

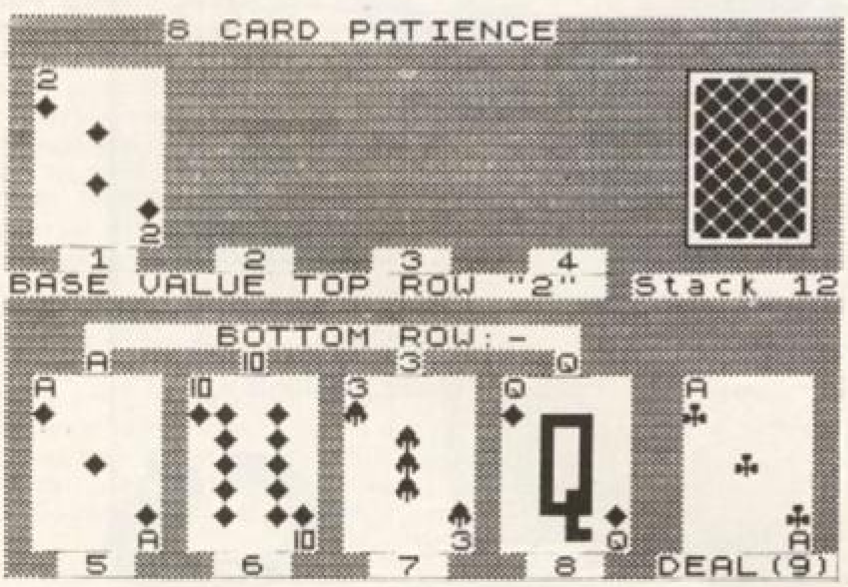


This is a traditional Patience game, which involves us in quite a lot of intricate programming. This is because we can have up to ten different stacks of cards displayed, plus another one 'in our hand', and we need a separate string for each stack so that we can know where each card is. These stacks have to moved visually from one place to another.

Fairly full rules are given in the programme. Briefly the object is to build four stacks by suit in ascending order, from a further hidden stack and the remainder in our hand. The card presentation is on a five by seven matrix which allows for a reasonable representation of a standard card in two rows of five. For identification purposes each card position is numbered underneath and it is these numbers that you enter to make a move. Also indicated is the base card i.e. the card at the bottom of each stack.

Programme operation

The fifty two cards are each represented by a string of five characters representing card name, value (2 characters), suit, and colour. These are originally formed in A$ which is then shuffled. After the hidden stack of 13 cards has been formed the remainder becomes P$ and this is manipulated to gradually build up the various stacks by dealing them out three at a time. The cards are not transferred to another string as they are dealt but a variable PSL slices P$ at the correct place. A choice of two deals is available, the second of which is a shuffling deal which gives an easier game. A string array R$ holds the eight possible stacks of played cards and is shadowed by a numerical array R which acts as a slicer for each stack. Also the hidden stack has a slice SLST.



Program lines

A line by line breakdown of the action is as follows:

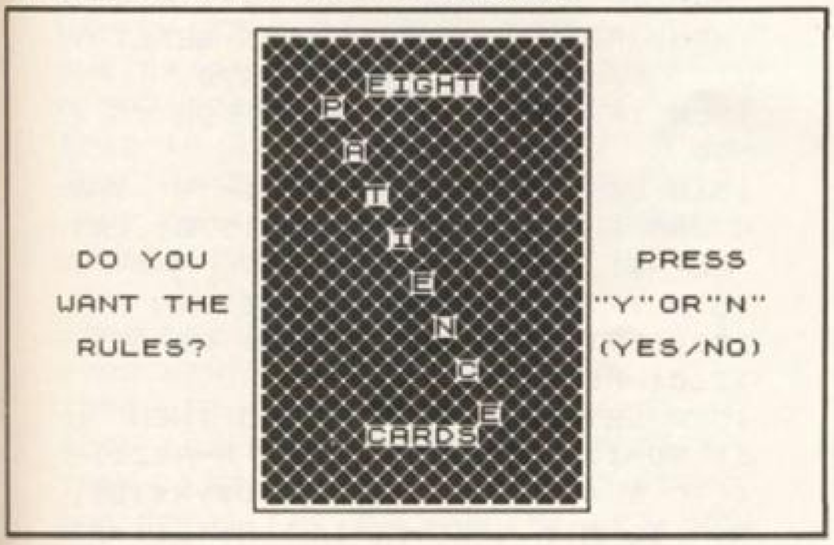
1020-1060	Initialises and calls up instructions
1070-1090	Chooses hidden stack
1100-1200	Deals out stack face down then removes the first card to top row.
1210-1230	Prints out bottom row of cards then moves on to 1420 which is start of main programme.
1250	Prints a card back
1260-1410	Prints out a card. The position is fixed by a defined function based on the card that has been chosen.
1420-1430	Choose to move a card or to deal.
1490-1530	Subroutine to accept a card move. Will reject an illegal entry.
1540-1560	Chooses appropriate routine from either moving a stack or taking a dealt card.
1570-1580	Reprints a card if entry is at fault
1585-1690	Moves cards one by one from one stack to another.
1700-1770	Checks validity of the destination. If wrong sets FLT glag. If correct prints card at new location.
1780-1800	Removes card from the pack and resets variable holding number of cards.
1810-1830	Removes card from one of the stacks and resets slicer.
1850-1890	Removes card from stack of hidden cards
1900-2050	Deals the cards. The correct deal will be used as set by DEALTYP
2060-2150	Sets up and shuffles the pack.
2160	Subroutine called at end of others to continue game.
2170-2260	End game. Called when the RES flag is set by programmer or player resigns. Gives options for continuing.
2270-2630	Instruction routine. It can be a temptation to miss out instructions but it is recommended you enter these for future reference. In any case the lines from 2530 must be included as they set the deal-type. The POKEs seen in this section POKE values directly into the attributes file to highlight certain points.
2640-2690	Initialise values.
2700-2880	Sets up the graphics
9990-9993	SAVEing and LOADing instructions.

Variables

A$	Complete pack
B$	Pictures of suits
C$	Card in play
D$	Temporary string
E$ F$	Components of card
K$	Base value for top row
T$	Title
P$	Pack in play
R$(x)	Stacks of cards in rows
S$	Spare stack
z$	Choice of cards to play
COL	Colour of card
DEALT	Cards dealt
DEALTYP	Type of deal to use
ENDP	Flag to indicate all cards used
FA FD	Temporary Flags
FLT	Flag to indicate faulty input
L M	Position of cards (face down)
B M N P Q	FOR-NEXT loops
PCD	Number of cards in pack
PSL	Slicer for pack
R(x)	Slicer for R$
RES	Flag to indicate game end
STACK	Cards in hidden stack
SLST	Slicer for stack
X Y	Position of card to be printed
Z(x)	Card to be played. Derived from Z$

NOTE

This program will not fit into a 16K machine.

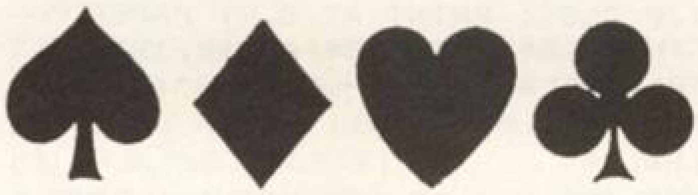



```
   1 REM ***********************
         *Underlined characters*
         *are entered in       *
         *GRAPHICS mode.       *
         ***********************
   2 REM U.D.GRAPHICS CHARACTERS
 IN  LINES 124Ø.131Ø.132Ø.135Ø.1
74Ø.182Ø
1ØØØ REM PATIENCE-C.N.GØØCH 1984
.
1Ø2Ø DEF FN A(X)=(PEEK X)+192: D
EF FN B(X)=(PEEK X)-192
1Ø3Ø DEF FN C(X)=6*(X=2 OR X=6)+
12*(X=3 OR X=7)+18*(X=4 OR X=8)
1Ø4Ø RANDOMIZE : RANDOMIZE
1Ø5Ø PAPER 4: CLS : LET K$="4"
1Ø6Ø GO SUB 264Ø: GO SUB 227Ø: G
O SUB 2Ø6Ø: BORDER 4: PAPER 4: I
NK Ø: CLS : PAUSE 5Ø
1Ø7Ø REM CHOOSE STACK
1Ø8Ø LET S$=A$( TO 65): LET P$=A
$(66 TO ): LET PSL=1
1Ø9Ø PAUSE 5Ø
11ØØ REM PRINT STACK
111Ø PRINT AT Ø,6; PAPER 6; INK
Ø;"8 CARD PATIENCE"
112Ø FOR N=1 TO 4: PRINT AT 9,6*
N-4; PAPER 6;" ";N;" ";AT 21,6*N
-4;" ";N+4;" ": NEXT N: PRINT AT
 21,25; PAPER 6;"DEAL(9)"
113Ø PAUSE 5Ø: INK 1: PRINT AT 2
,26; PAPER 7;"GHHHI": FOR N=3 TO
 7: PRINT AT N,26; PAPER 7;"JEEE
N": NEXT N: PRINT AT 8,26; PAPER
 7;"KLLLM"
114Ø INK 1: FOR M=1 TO 13: FOR N
=2 TO 7: PRINT AT N,26; PAPER 7;
"JEEEN": PRINT AT N+1,26; PAPER
7;"KLLLM": NEXT N: PRINT AT 2,26
; PAPER 7;"GHHHI": BEEP .Ø1,2: N
EXT M: PRINT AT 1Ø,24; PAPER 6;"
Stack ";STACK
115Ø PRINT  PAPER 4;AT 11,21;"
        ": IF ENDP THEN  PRINT A
T 11,21; PAPER 6;"PACK USED"
116Ø LET L=14: INK 1: FOR M=1 TO
 19 STEP 6: GO SUB 125Ø
117Ø NEXT M
118Ø REM CARD OFF STACK TO TOP
119Ø GO SUB 216Ø: LET X=Ø: LET Y
=2: LET C$=S$(SLST TO SLST+4): L
ET SLST=SLST+5: LET K$=C$(1): LE
T R$(1)=C$: LET R(1)=6: LET STAC
K=STACK-1: PRINT AT 1Ø,3Ø; PAPER
 6;"  ";AT 1Ø,3Ø;STACK: GO SUB 1
26Ø
12ØØ PRINT AT 1Ø,Ø; INK Ø; PAPER
 5;"BASE VALUE TOP ROW """;K$;""
"";"  ";AT 12,3; PAPER 5;"      BO
TTOM ROW:-  "
121Ø REM PRINT INITIAL BOTTOM R.
122Ø GO SUB 216Ø: LET X=Ø: LET F
A=5: FOR N=Ø TO 19 STEP 5: LET C
$=P$(1+N TO 5+N): LET R$(FA, TO
5)=C$: LET R(FA)=6: LET FA=FA+1:
 LET Y=14: GO SUB 126Ø: PRINT AT
 13,FN C(FA-1)+3; PAPER 5;C$(1):
 LET X=X+6: NEXT N: LET P$=P$(21
 TO ): LET PCD=35: LET CD1=Ø: LE
T DEAL=1
123Ø GO TO 142Ø
124Ø REM  PRINT CARD BACK
125Ø INK 1: PRINT AT L,M; PAPER
7;"GHHHI": FOR N=L+1 TO L+5: PRI
NT AT N,M; PAPER 7;"JEEEN": NEXT
 N: PRINT AT L+6,M; PAPER 7;"KLL
LM": RETURN
126Ø REM PRINT CARD
```

```
1270 PRINT  PAPER 4;AT 12,25;"
     ";AT 13,25;"        ": IF EN
DP THEN  PRINT AT 12,25; PAPER 6
;" END OF";AT 13,25;" PACK  "
1280 INK VAL C$(5): LET E$=C$(4)
: LET F$=C$(1): IF C$(1)=K$ THEN
 FOR B=1 TO 5: BEEP .01,5*B: NE
XT B
1290 PAPER 7: FOR M=Y TO 8+12*(Y
>3): PRINT AT M,X+1; INK 0;"
 ": NEXT M: PRINT AT Y,X+1;F$;AT
 Y+6,X+5;F$
1300 PRINT AT Y+1,X+1;E$;AT Y+5,
X+5;E$
1310 IF (F$="A")+(F$="3")+(F$="5
")+(F$="9") THEN  PRINT AT Y+3,X
+3;E$
1320 IF (F$="2")+(F$="3") THEN
PRINTATY+2,X+3;E$;AT Y+4,X+3;E$
1330 IF (F$="4")+(F$="5")+(F$="8
")+(F$="9")+(F$="A") THEN  PRINT
 AT Y+2,X+2;E$;AT Y+2,X+4;E$;AT
Y+4,X+2;E$;AT Y+4,X+4;E$
1340 IF (F$="6")+(F$="7") THEN
PRINT AT Y+1,X+2;E$;AT Y+1,X+4;E
$;AT Y+3,X+2;E$;AT Y+3,X+4;E$;AT
 Y+5,X+2;E$;AT Y+5,X+4;E$
1350 IF (F$="7") THEN  PRINT AT
Y+3,X+3;E$
1360 IF (F$="8")+(F$="9")+(F$="A
") THEN  PRINT AT Y+1,X+2;E$;AT
Y+1,X+4;E$;AT Y+5,X+2;E$;AT Y+5,
X+4;E$
1370 IF (F$="A") THEN  PRINT AT
Y+3,X+2;E$;AT Y+3,X+4;E$
1380 IF F$="J" THEN  PRINT AT Y+
1,X+3;"▄▄";AT Y+2,X+4;"▌";AT Y+3
,X+4;"▌";AT Y+4,X+2;"▌▌";AT Y+5
,X+2;"▀▀"
1390 IF F$="Q" THEN  PRINT AT Y+
1,X+2;"▄▄";AT Y+2,X+2;"▌▌";AT
Y+3,X+2;"▌▌";AT Y+4,X+2;"▌▟";A
T Y+5,X+2;"▀▛";AT Y+6,X+3;"▀"
1400 IF F$="K" THEN  PRINT AT Y+
1,X+2;"▌█";AT Y+2,X+2;"▌█";AT Y
+3,X+2;"█";AT Y+4,X+2;"▌█";AT Y
+5,X+2;"▌█"
1410 RETURN
1420 REM ELECT TO MOVE
1430 IF NOT STACK AND NOT PCD AN
D R(5)+R(6)+R(7)+R(8)=4 THEN  LE
T RES=0: GO TO 2180
1440 PRINT #1;AT 0,0;"  PRESS(1)
MOVE CARD (2)DEAL      ";AT 1,0;"
 OR (0)RESIGN    ": PAUSE 0
1450 IF INKEY$="1" THEN  GO TO 1
490
1460 IF INKEY$="2" THEN  GO TO 1
900
1470 IF INKEY$="0" THEN  LET RES
=1: INPUT "": GO TO 2180
1480 PAUSE 0: GO TO 1450
1490 INPUT ;AT 0,0;"ENTER TWO NU
MBERS TOGETHER. CARD";AT 1,0;"TO
 MOVE & DESTINATION.eg.95"; LINE
 Z$
1500 IF Z$="" OR LEN Z$<2 OR LEN
 Z$>2 THEN  GO TO 1490
1510 IF (CODE Z$(1)<49 OR CODE Z
$(2)<49)+(CODE Z$(1)>57 OR CODE
Z$(2)>57) THEN  INPUT "ENTRY UNA
CCEPTABLE.RE-ENTER"; LINE Z$: GO
 TO 1500
1520 LET Z1=VAL Z$(1): LET Z2=VA
L Z$(2)
1530 IF Z1<5 OR Z1=Z2 OR Z2=9 TH
EN  INPUT "ENTRY UNACCEPTABLE.RE
-ENTER"; LINE Z$: GO TO 1500
1540 IF NOT DEALT AND Z$(1)="9"
THEN  PRINT #1;"NO CARDS DEALT Y
ET": PAUSE 100: GO TO 1490
1550 IF Z$(1)<>"9" THEN  GO TO 1
590
1560 LET FLT=0: IF z$(1)="9" THE
N  GO SUB 1670: GO SUB 1570: LET
 FLT=0: GO TO 1430
1570 IF NOT FLT THEN  RETURN
1580 LET Y=14: LET X=25: GO SUB
1260: RETURN
1590 LET FLT=0: IF Z2>=5 THEN  L
ET FD=1: LET D$="": FOR M=R(Z1)-
1 TO 1 STEP -5: LET D$=D$+R$(Z1,
M-4 TO M): NEXT M: LET R$(Z1)=D$
1600 LET FLT=0: FOR N=R(Z1) TO 6
 STEP -5
1610 IF N=6 OR FD THEN  FOR M=1
TO 7: PRINT AT 13+M,1+FN C(Z1);
PAPER 4;"       ": NEXT M: GO TO 1
630
1620 LET C$=R$(Z1,R(Z1)-10 TO R(
Z1)-6): LET Y=14: LET X=FN C(Z1)
: GO SUB 1260
1630 LET C$=R$(Z1,R(Z1)-5 TO R(Z
1)-1): LET Y=2+12*(Z2>=5): LET X
=FN C(Z2): GO SUB 1700
1640 IF FLT THEN  LET C$=R$((Z1)
,R(Z1)-5 TO R(Z1)-1): LET Y=14:
LET X=FN C(Z1): GO SUB 1260: GO
TO 1430
1650 GO SUB 1810
1660 NEXT N: LET FD=0: GO TO 143
0
```

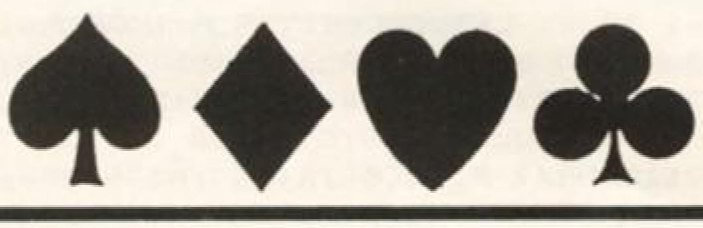
```
1670 IF PSL=1 THEN  FOR N=14 TO
20: PRINT AT N,26; PAPER 4;"
 ": NEXT N: GO TO 1690
1680 LET C$=P$(PSL-5 TO PSL-1):
LET Y=14: LET X=25: GO SUB 1260
1690 LET C$=P$(PSL TO PSL+4): LE
T Y=2+12*(Z2>4): LET X=FN C(Z2)
1700 IF R(Z2)=1 AND C$(1)=K$ THE
N  GO SUB 1260: GO SUB 1790: RET
URN
1710 IF R(Z2)=1 AND R$(Z1,1)=K$
THEN  GO SUB 1260: GO SUB 1790:
RETURN
1720 IF R(Z2)=1 THEN  LET FLT=1:
 RETURN
1730 IF Z2<5 AND VAL C$(2 TO 3)=
1 AND VAL R$(Z2,R(Z2)-4 TO R(Z2)
-3)=13 AND C$(4)=R$(Z2,R(Z2)-2)
THEN  GO SUB 1260: GO SUB 1790:
RETURN
1740 IF Z2>=5 AND VAL C$(2 TO 3)
=13 AND VAL R$(Z2,R(Z2)-4 TO R(Z
2)-3)=1 AND C$(4)=R$(Z2,R(Z2)-2)
 THEN  GO SUB 1260: GO SUB 1790:
 RETURN
1750 IF Z2<5 AND VAL C$(2 TO 3)=
VAL R$(Z2,R(Z2)-4 TO R(Z2)-3)+1
AND C$(4)=R$(Z2,R(Z2)-2) THEN  G
O SUB 1260: GO SUB 1790: RETURN
1760 IF Z2>=5 AND VAL C$(2 TO 3)
=VAL R$(Z2,R(Z2)-4 TO R(Z2)-3)-1
 AND C$(4)=R$(Z2,R(Z2)-2) THEN
GO SUB 1260: GO SUB 1790: RETURN

1770 LET FLT=1: RETURN
1780 REM MOVE CARD;DEC PACK
1790 LET R$(Z2,R(Z2) TO R(Z2)+4)
=C$: LET R(Z2)=R(Z2)+5: IF Z1=9
THEN  LET P$=P$( TO PSL-1)+P$(PS
L+5 TO ): LET PSL=PSL-5: LET PCD
=PCD-1: RETURN
1800 RETURN
1810 REM REMOVE CARD FROM TRICK
1820 LET R(Z1)=R(Z1)-5: IF R(Z1)
<=1 THEN  GO TO 1850
1830 LET R$(Z1)=R$(Z1, TO (R(Z1)
-1)): RETURN
1840 REM CARD OFF STACK
1850 IF INKEY$<>"" THEN  GO TO 1
850
1860 IF NOT STACK  THEN  RETURN
1870 LET STACK=STACK-1: LET C$=S
$(SLST TO SLST+4): LET SLST=SLST
+5: LET R$(Z1)=C$: LET R(Z1)=6:
LET Y=14: LET X=FN C(Z1)
1880 PRINT #1;AT 0,0;"PRESS ""EN
TER"" TO FETCH NEW";AT 1,0;"CARD
 FROM STACK": PAUSE 0: IF STACK=
0 THEN  FOR N=2 TO 8: PRINT AT N
```



```
,26; PAPER 4;"     ": NEXT N:
1890 GO SUB 1260: PRINT AT 10,30
; PAPER 6; INK 0;"  ";AT 10,30;S
TACK;AT 13,FN C(Z1)+3; PAPER 5;C
$(1): RETURN
1900 REM DEAL
1910 IF INKEY$<>"" THEN  GO TO 1
910
1920 IF PCD=0 THEN  PRINT AT 14,
27; PAPER 7; INK 2;"NO";AT 16,26
;"CARDS";AT 19,26;"LEFT": GO TO
1430
1930 LET DEALT=DEALT+1:
1940 IF ENDP THEN  LET PSL=1: LE
T ENDP=0
1950 LET PSL=PSL+15-5*(PSL=1)
1960 IF PSL>=PCD*5 OR PSL=PCD*5-
4 THEN  LET PSL=PCD*5-4: LET END
P=1
1970 IF DEALTYP=2 AND PSL=PCD*5-
14 THEN  LET ENDP=1
1980 IF DEALTYP=2 THEN  GO SUB 2
000
1990 LET C$=P$(PSL TO PSL+4): LE
T Y=14: LET X=25: GO SUB 1260: G
O TO 1430
2000 IF PCD=3 THEN  LET P$=P$(PS
L TO PSL+4)+P$(PSL-5 TO PSL-1)+P
$(PSL-10 TO PSL-6): RETURN
2010 IF PCD=2 THEN  LET P$=P$(PS
L TO PSL+4)+P$(PSL-5 TO PSL-1):
RETURN
2020 IF PCD=1 THEN  RETURN
2030 LET D$=P$(PSL-10 TO PSL+4)
2040 LET P$(PSL-10 TO PSL-6)=D$(
11 TO 15): LET P$(PSL TO PSL+4)=
D$(1 TO 5)
2050 RETURN
2060 REM SET UPPACK
2070 LET A$="": LET D$="A0120230
34045056067078089098103J11Q12K13"
2080 PRINT AT 8,9; INK 1; PAPER
5;B$;B$;B$;B$;AT 10,10; FLASH 1;
 PAPER 7; INK 2;"PREPARING PACK"
;AT 12,9; FLASH 0; PAPER 5; INK
1;B$;B$;B$;B$
2090 FOR Q=1 TO 39 STEP 3: FOR P
=1 TO 4: LET COL=0: IF P=1 OR P=
3 THEN  LET COL=2
2100 LET A$=A$+D$(Q TO Q+2)+B$(P
)+STR$ COL
2110 NEXT P: NEXT Q
```

```
2120 CLS : PRINT AT 8,9; PAPER 5
; INK 1;B$;B$;B$;B$;AT 10,12; PA
PER 7; FLASH 1;"SHUFFLING";AT 12
,9; PAPER 5; INK 1; FLASH 0;B$;B
$;B$;B$
2130 FOR N=1 TO 60: LET A=1+(5*(
1+INT (RND*50))): IF n/2=INT (n/
2) THEN  LET A$=A$(A TO A+4)+A$(
 TO A-1)+A$(A+5 TO ): BEEP .01,N
2140 IF n/2<>INT (n/2) THEN  LET
 A$=A$( TO A-1)+A$(A+5 TO )+A$(A
 TO A+4): BEEP .01,N+10
2150 NEXT N: CLS : RETURN
2160 INPUT "PRESS ENTER TO CONTI
NUE";LINE Z$: BEEP.05,10: RETURN
2170 REM END GAME
2180 PAPER 5: INK 1: FOR N=12 TO
 21: PRINT AT N,0; PAPER 5;"
                               ": N
EXT N
2190 IF RES THEN  PRINT AT 13,1;
"BAD LUCK: THE CARDS DID NOT ";A
T 14,1;"RUN YOUR WAY"
2200 IF NOT RES THEN  PRINT AT 1
3,1; FLASH 1;"!!!!! WELL DONE !!
!!!!";AT 14,1; FLASH 0;"YOUR PAT
IENCE IS REWARDED"
2210 PRINT AT 16,3;"YOU MAY";AT
17,5;"1)PLAY AGAIN:SAME DEAL";AT
 18,5;"2)PLAY AGAIN:OTHER DEAL";
AT 19,5;"3)FINISH PLAYING"
2220 PAUSE 0
2230 IF INKEY$="1" THEN  BORDER
4: PAPER 4: INK 0: CLS : GO SUB
2640: GO SUB 2120: GO TO 1070
2240 IF INKEY$="2" THEN  LET DEA
LTYP=DEALTYP+1-2*(DEALTYP=2): BO
RDER 4: PAPER 4: INK 0: CLS : GO
 SUB 2640: GO SUB 2120: GO TO 10
70
2250 IF INKEY$="3" THEN  CLS : P
RINT AT 10,0;"START TAPE TO LOAD
 NEXT GAME": LOAD ""
2260 PAUSE 0: GO TO 2230
2270 REM INSTRUCTIONS
2280 PAPER 2: INK 7: BORDER 5: C
LS
2290 PRINT AT 0,9; INK 1; PAPER
7;"GHHHHHHHHHHHHHI": FOR N=1 TO
20: BEEP .01,2*N: PRINT AT N,9;
INK 1; PAPER 7;"JEEEEEEEEEEEEEN"
: NEXT N: PRINT AT 21,9; INK 1;
PAPER 7;"KLLLLLLLLLLLLLM"
```



```
2300 PRINT AT 2,14; BRIGHT 1; PA
PER 6; INK 0;"EIGHT";AT 18,14;"C
ARDS"
2310 LET T$="PATIENCE":: FOR M=2
 TO 17 STEP 2: BEEP .01,20-M: PR
INT AT M+1,11+M/2; PAPER 6; BRIG
HT 1; INK 0;T$(M/2): NEXT M
2320 PRINT AT 10,1; BRIGHT 1;"DO
 YOU";AT 12,0;"WANT THE";AT 14,1
;"RULES?";AT 10,25;" PRESS";AT 1
2,24;"""Y""OR""N""";AT 14,24;"(Y
ES/NO)"
2330 PAUSE 0: IF INKEY$="N" OR I
NKEY$="n" THEN  FOR n=10 TO 20:
BEEP n/100,20: NEXT n: CLS : GO
SUB 2620: GO TO 2530
2340 IF INKEY$<>"Y" AND INKEY$<>
"y" THEN  GO TO 2330
2350 PAPER 4: INK 0: CLS : PRINT
 AT 0,4; PAPER 5;"RULES FOR 8 CA
RD PATIENCE"
2360 GO SUB 2620
2370 PRINT AT 2,8; PAPER 7;"The
playing area";AT 3,8;"is set out
 thus-";AT 4,8; FLASH 1;"*";AT 4
,9; FLASH 0;"A Spare Stack";AT 5
,13;"of 13 cards": LET L=2: LET
M=26: GO SUB 1240: POKE 22716,FN
 A(22716)
2380 GO SUB 2160: POKE 22716,FN
B(22716): PRINT AT 4,8; PAPER 7;
" "
2390  PRINT AT 6,8; INK 0; PAPER
 7; FLASH 1;"*";AT 6,9; FLASH 0;
"A  bottom row": LET L=14: FOR M
=1 TO 20 STEP 6: GO SUB 1240: PO
KE 23074+M,FN A(23074+M): NEXT M
2400 GO SUB 2160: FOR M=1 TO 20
STEP 6: POKE 23074+M,FN B(23074+
M): NEXT M: PRINT AT 6,8; PAPER
7;" "
2410 PRINT AT 7,8; PAPER 7; INK
0; FLASH 1;"*";AT 7,9; FLASH 0;"
A Top card": LET C$="404D0": LET
 X=0: LET Y=2: GO SUB 1260
2420 POKE 22691,FN A(22691): GO
SUB 2160: PRINT AT 7,8; PAPER 7;
" ": POKE 22691,FN B(22691)
2430 PRINT AT 8,8; PAPER 7; INK
0; FLASH 1;"*";AT 8,9; FLASH 0;"
An area where ";AT 9,9;"cards ma
y be ";AT 10,18;"dealt"
2440 FOR N=14 TO 20: PRINT AT N,
26;"      ": NEXT N: POKE 23100,F
N A(23100)
2450 GO SUB 2160: PRINT AT 8,8;
PAPER 7;" ": POKE 23100,FN B(231
00): GO SUB 2620
2460 PRINT AT 2,8; PAPER 7;"Card
```

```
s are dealt";AT 3,8;"three at a
time";AT 4,8;"The top row is";AT
 5,8;"completed with";AT 6,8;"ca
rds the same";AT 7,8;"value as t
he";AT 8,8;"first from the ";AT
9,8;"dealt pack or  ";AT 10,8;"b
ottom row"
2470 GO SUB 2160: GO SUB 2620
2480 PRINT AT 2,8;"Cards from th
e";AT 3,8;"bottom row, or";AT 4,
8;"dealt pile are";AT 5,8;"built
 in ascend-";AT 6,8;"ing order o
n the";AT 7,8;"top row and in";A
T 8,8;"descending on";AT 9,8;"th
e bottom"
2490 GO SUB 2160: GO SUB 2620
2500 IF INKEY$<>"" THEN  GO TO 2
500
2510 PRINT AT 2,8;"When cards ar
e";AT 3,8;"moved from the";AT 4,
8;"bottom to top";AT 5,8;"row or
 along";AT 6,8;" bottom the gap"
;AT 7,8;"is filled from";AT 8,8;
"the spare stack.";AT 9,8;"A ";
INK 5;"■"; INK 0;" marker shows"
;AT 10,8;"base value": PRINT AT
13,9; INK 5; FLASH 1;"■"
2520 GO SUB 2160: PRINT AT 13,9;
 PAPER 4;" ": GO SUB 2620
2530 PRINT  PAPER 7; INK 0;AT 2,
8;"You must choose";AT 3,8;"the
type of deal";AT 4,8;"1)Straight
 block";AT 5,10;"of three cards"
2540 PRINT  PAPER 7; INK 0;AT 6,
8;"2)Block of three";AT 7,10;"in
 reversed";AT 8,10;"order";AT 9,
10;"""1"" is harder";AT 10,10;"g
ame": PRINT #1;AT 1,0; PAPER 7;
INK 0;"PRESS ""1"" OR ""2"""
2550 PAUSE 0: INPUT "": IF  INKE
Y$="1" THEN  LET DEALTYP=1
2560 IF  INKEY$="2" THEN  LET DE
ALTYP=2
2570 IF INKEY$<>"1" AND INKEY$<>
"2" THEN  GO TO 2530
2580 IF INKEY$<>"" THEN  GO TO 2
580
2590 GO SUB 2620: PRINT AT 2,8;
PAPER 7; INK 0;"To repeat rules"
;AT 3,8;"press ""1""";AT 5,8;"ot
herwise press";AT 6,8;"any key t
o play"
2600 PAUSE 0: IF INKEY$="1" THEN
  GO TO 2350
2610 CLS : RETURN
2620 FOR N=2 TO 10: PRINT AT N,8
; PAPER 7;"                 ": NE
XT N: RETURN
2630 RETURN
2640 REM INIT
2650 LET A=1: LET SL=1: LET STAC
K=13
2660 LET FD=0: LET ENDP=0: LET D
EALT=0: LET SLST=1
2670 DIM R$(9,85): DIM R(8): FOR
 N=1 TO 8: LET R(N)=1: NEXT N
2680 LET B$="BDCE"
2690 RETURN
2700 REM GRAPHICS
2710 BORDER 2: CLS : PRINT AT 10
,2;"STOP THE TAPE PLEASE"
2720 RESTORE 2700: FOR N=0 TO 14
: FOR M=0 TO 7: BORDER M: BEEP
01,N
2730 READ A: POKE USR CHR$ (97+N
)+M,A: NEXT M: NEXT N: GO TO 105
0
2740 DATA 0,94,82,82,82,82,94,0
2750 DATA 0,16,56,124,254,124,56
,16
2760 DATA 0,108,254,254,124,124,
56,16
2770 DATA 16,56,124,124,254,214,
84,16
2780 DATA 0,56,56,16,214,254,214
,16
2790 DATA 126,189,219,231,231,21
9,189,126
2800 DATA 255,128,128,135,135,15
5,157,158
2810 DATA 255,0,0,231,231,219,18
9,126
2820 DATA 255,1,1,225,225,217,18
5,121
2830 DATA 158,157,155,135,135,15
5,157,158
2840 DATA 158,157,155,135,135,12
8,128,255
2850 DATA 126,189,219,231,231,0,
0,255
2860 DATA 121,185,217,225,225,1,
1,255
2870 DATA 121,185,217,225,225,21
7,185,121
2880 DATA 128,128,128,128,128,12
8,128,128
9990 SAVE "patience" LINE 2700
9992 CLS : PRINT "SWITCH PLUGS A
ND REWIND TO ""VERIFY.   START
TAPE"
9993 VERIFY "patience": CLS : PR
INT "ALL OK"
```



HIRE SPECTRUM SOFTWARE

OVER 280 different tapes for hire including **ARCADE, ADVENTURE, BUSINESS, EDUCATIONAL**, etc. –even an m/c Compiler. **FREE** 26 page catalogue. **FREE** newsletter, with hints, tips and our top tape chart.

OVER 3000 tapes in stock, with up to 60 copies of each title for fast service. All tapes sent by 1st class post.

LOWEST PRICES – hire up to 3 tapes at a time, from 63p each for two weeks hire. (Plus p&p and VAT). European members welcome.

Tapes for sale at DISCOUNT prices. Telephone 01-661 9240 (9am - 5pm) or write for further details, or complete the coupon and **JOIN TODAY** – you've nothing to lose!

SAVE £3.00!!

For a limited period, we are offering **HALF-PRICE** membership to readers of this magazine. Join now, **LIFE** membership is only £3.00 (normally £6.00).

SWOP SHOP

A new service from N.S.L. **SWOP** your unwanted tapes with tapes from other members for a small handling charge of only 60p (plus p&p and VAT)

NATIONAL SOFTWARE LIBRARY

200 Mulgrave Road, Cheam, Surrey SM2 6JT.

I enclose cheque/postal order for £3.00 for LIFE membership. Please rush my membership kit to me. If, within 28 days, I'm not delighted with your service you'll refund my membership fee..

Name ____________________

Address ____________________

____

____________________ ZXC4

FLOPPY DISK SYSTEM FOR SPECTRUM

FLOPPY DISK SYSTEM FOR SPECTRUM

Our new model - BETA DISK is even better than ever.
Some of the main features:

- ★ Disk Operating System in EPROM
- ★ Uses ONLY 128 bytes of Spectrum Memory
- ★ Locates below BASIC programs
- ★ Uses Spectrum Keywords
- ★ Supports up to four disk drives
- ★ Compatible with 40, 80 single or double sided drives
- ★ Up to 1.6M bytes storage
- ★ Random access is provided
- ★ BASIC programs can be merged
- ★ Duplicated Spectrum connector is provided
- ★ Password protected.

£85.00 ex VAT
£2.00 p&p

Technology Research Limited
Unit 18 Central Trading Estate,
Staines, Middlesex
Tel: Staines (0784) 63547

SOFTWARE ON sinclair ZX-Spectrum



LOGO *The new educational language with Turtle graphics. Logo is an excellent tool for practically exploring designs and patterns giving experience of lengths, directions, angles, radii and colour*
Runs on a 48K Spectrum **£9.95** inc. VAT

Kuma Computers Ltd

HOMEBUDGET — *Helps you save money by planning your cash for the year ahead then displaying your accounts on useful bar charts utilising spectacular 3D graphics. A professional program for those who like to plan ahead.* **£6.95 inc. VAT**

See us on PRESTEL ✱ 2473222 #

SUPERCHESS 3.0 — *The strongest chess program yet for the Spectrum Computer. Unlike other programs it employs artificial intelligence techniques to allow deeper searching along the main lines of play.* **£8.95 inc. VAT**

COMET: KUMA   TELECOM GOLD: KUM 001

MAP OF U.K. — *A large scale 75 x 60 mile scrolling map is now available on the 48K Spectrum. It is easy to use and extremely fast. The program also includes a game using a drifting hot air balloon. Educational and entertaining.* **£11.95 inc. VAT**

Telex 849462 TELFAC KUMA

HISOFT PASCAL — *Virtually a full implementation of the language allowing the user to develop true high-level language skills while attaining execution speed close to that of machine code. Complete with 30 page manual.* **£25.00 inc. VAT**

Unit 12, Horseshoe Park, Horseshoe Road, Pangbourne, Berks RG8 7JW

ZEN EDITOR/ASSEMBLER — *Zen enables you to write, assemble, edit and run programs in Z80 Assembly language. These operate faster and take up considerably less memory space than Basic programs. A must for the serious beginner.* **£12.50 inc. VAT**

HORSE RACE FORECAST — *Uses past and present races and information to forecast the chances of a horse winning. Can you afford to be without this program.* **£15.00 inc. VAT**

and many more send for catalogue

PHONE NOW (073 57) 4335
or SEND FOR DETAILS

Please tick box ☑ for information required, and send coupon to:

Kuma Computers Ltd.
12, Horseshoe Park
Horseshoe Road, Pangbourne
Berks. RG8 7JW

Hardware	Software
☐ SHARP PC-5000	☐
☐ SIRIUS	☐
☐ SHARP PC 1500	☐
☐ SHARP 700	☐
☐ SHARP A	☐
☐ SHARP 3541	☐
☐ SAGE 68000 System	☐
☐ APRICOT	☐
☐ EPSON HX-20	☐
☐ EPSON QX-10	☐
☐ NEWBRAIN	☐
☐ SPECTRUM	☐
☐ ELECTRONIC MAIL	☐
☐ PRINTERS	☐
SPECTRUM Software	☐

NAME ____________________

ADDRESS ____________________

POST CODE ____________________

Port-ability

An Ambitious project for the adventurous user written by M P Moore of Petron Electronics.

Have you ever wanted your computer to actually, physically DO something? Here is a special article for those who would like to experiment with controlling things with his machine. Cleverly written so that it is compatible with both ZX81 and Spectrum, this first article shows how to make a versatile unit in simple terms so that even the least experienced can have a go. The next article will develop and create something to use it on. A great way of utilising that old ZX81 stored in the loft and an exciting extension to your machine.

This circuit was designed to allow up to twelve peripherals to be connected to a ZX81 or Spectrum. The interface will work with the ZX81 and Spectrum with no modifications to the circuit. No knowledge of electronics is required to build it although it is assumed that the constructor has access to a fine tipped soldering iron.

The interface was designed to allow a diverse range of peripherals to be connected to the computer such as serial and parallel ports, digital to analogue and analogue to digital converters, printers, joysticks etc.

Four fourteen pin sockets are mounted on the printed circuit board (PCB), each socket can be connected to up to three devices, or two if handshake control signals are required in which case the third port would be used for handshake signals to and from the other two devices. Fourteen way DIP jumper cables are used to keep the interface neat and to give a professional finish.

Data lines D0 to D7, $\overline{READ}$, $\overline{WRITE}$, VCC (+5V), GND (OV) and two port enable lines ($\overline{PE}$) are taken to each socket (see fig. 1). The interface assigns three numbers to each socket (see table 1).

Each socket has two $\overline{PE}$ lines taken to it, these lines can be either 0 or 1. When one or both lines are at logic 0 a peripheral is being addressed, each combination of 00, 01, 10 will select a peripheral device while 11 will deselect.

Device Select

A peripheral device is selected by outputting its device number to port 65407. Data can then be read from or written to the peripheral using port 65471. Peripherals can be accessed using BASIC on the Spectrum, for example:

OUT 65407, 254

Selects device number 254

LET A = IN 65471

Reads data from the selected peripheral (in this case 254) to variable a

OUT 65471, A

Outputs the contents of variable A to the peripheral

ZX81 programming is slightly more complicated. Three short machine code routines are used (see table 2). These are located at line 1 REM X (where X is memory address 16514).

To select a peripheral device on the ZX81,

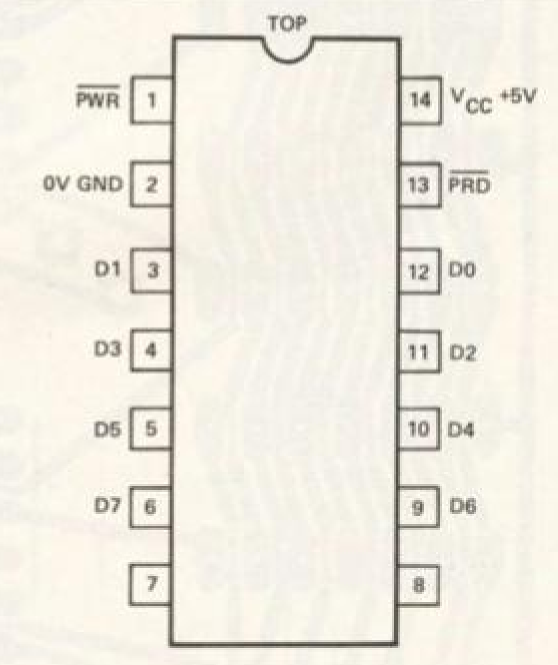
Socket	$\overline{PE}$	Device No.	Hex.
1	00	252	FC
1	01	253	FD
1	10	254	FE
2	00	243	F3
2	01	247	F7
2	10	251	FB
3	00	63	3F
3	01	127	7F
3	10	191	BF
4	00	207	CF
4	01	223	DF
4	10	239	EF
DESELECT	11	255	FF

Table 1 Peripheral device select numbers

```
1    REM XXXXXXXXXXXXXXXX (reserve 16
     bytes)
10   FOR F=16514 TO 16529
20   INPUT N
30   POKE F,N
40   NEXT F
```

Address	Data	Comments
16514	62	LD A,n entry point for device select.
16515	00	POKEd (device No.).
16516	211	OUT (n),A.
16517	127	n is device select port.
16518	201	RET.
16519	62	LD A,n entry point for data output.
16520	00	POKEd (data to be output).
16521	211	OUT (n),A.
16522	191	n is data I/O port.
16523	201	RET.
16524	6	LD B,n entry point for data input.
16525	0	n=0 (clear B).
16526	219	IN A,(n).
16527	191	n is data I/O port.
16528	79	LD C,A BC equals #00dd where dd is the data input.
16529	201	RET.

Table 2 Machine-code subroutines for the ZX81



SKT	PIN 7	PIN 8
1	$\overline{PE1}$	$\overline{PE0}$
2	$\overline{PE2}$	$\overline{PE3}$
3	$\overline{PE6}$	$\overline{PE7}$
4	$\overline{PE5}$	$\overline{PE4}$

Fig. 1 I/O socket connections

POKE 16515,n (where n is the device number)

RAND USR 16514 (and the peripheral is selected)

To output data to the selected peripheral, use

POKE 16520,d (where d is the data to be output or a variable name)

RAND USR 16519 (the data POKEd to 16520 will be output to the peripheral).

To input data use

LET A=USR 16524

In this case, data is input from the peripheral and placed in variable A.

The electronics — how it works

See fig. 2a. Diodes D1 and D2 in conjunction with resistor R1 form an OR gate whose output will only be at logic 0 if a write is being made to port 65407. These components are necessary since IORQ will go to logic 0 (in conjunction with M1 which is not used by this circuit) whenever the CPU recognises an interrupt. Since neither RD nor WR will be at logic 0 during any interrupt response, this OR gate ensures that there will be no spurious interference with the interface caused by the CPU interrupt system. IC1d ORs this

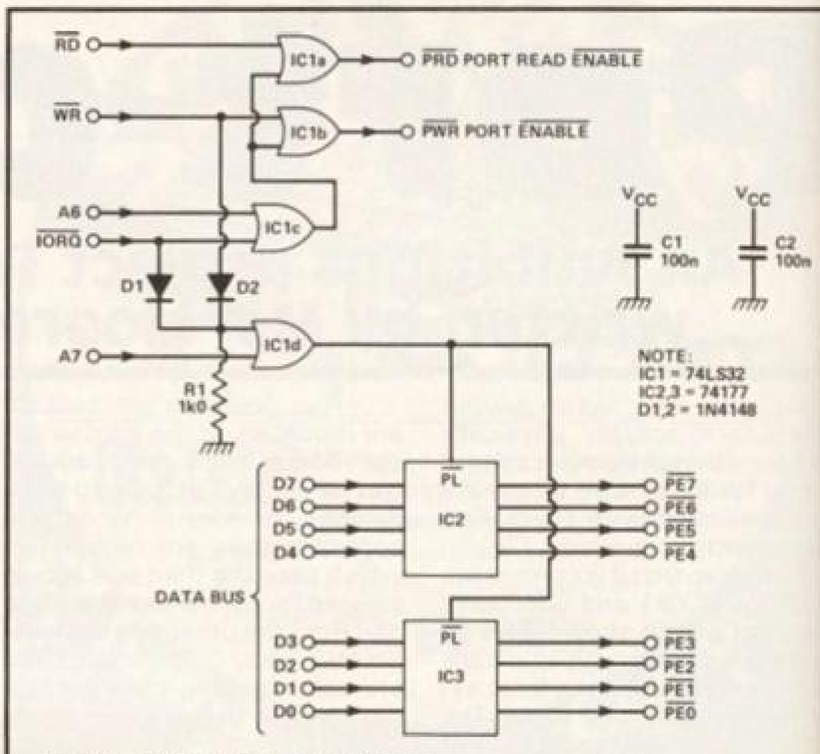


Fig. 2a Interface function diagram

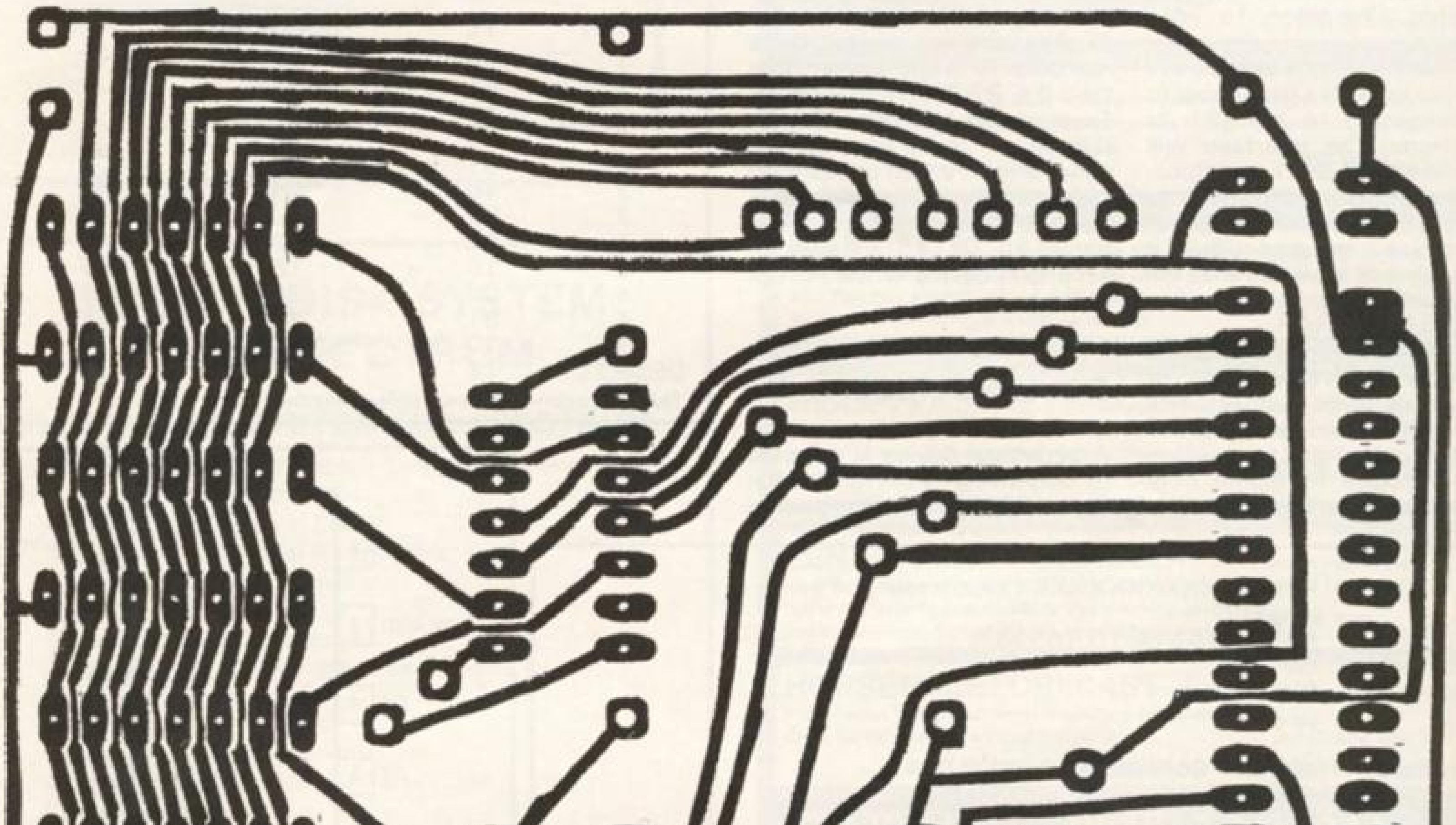


Fig. 2b PCB foil pattern

output with address line A7, the 'device select' address line. When $\overline{IORQ}$, $\overline{WR}$ and A7 are low (at logic 0) ICs 2 and 3 latch the data on their inputs and hold it on lines $\overline{PE}$ 0 to $\overline{PE}$ 7. (The 74177 latches data on a PL low to high state transition).

A6 is the address line used for data I/O. In conjunction with $\overline{IORQ}$ and $\overline{RD}$ or $\overline{WR}$, ICs 1a and 1b supply port read ($\overline{PRD}$) and port write ($\overline{PWR}$) lines; these will always be at logic 1 except when a read or write to a peripheral is intended.

A parallel interface is easily constructed using this circuit. To write to an external device all that is required is to OR the appropriate $\overline{PE}$ line with $\overline{PWR}$, data would be strobed into the peripheral upon a low to high or high to low transition of the output of this gate.

Inputing data is just as simple. Again an OR gate is used to OR a $\overline{PE}$ line (possibly the same one) with $\overline{PRD}$, the output of this gate can be taken straight to the $\overline{OE}$ (output enable) lines of a tri-state buffer, eg. 81LS97. Upon a read occuring, the data on the 81LS97s inputs would be read into the CPU.

Construction

See Fig. 3. You will need a soldering iron with a fine bit and fine solder (preferably multicore solder).

All components for the interface are mounted on a single sided printed circuit board (PCB); this has a number of tracks which pass between IC pins. Great care should be taken that no solder forms 'bridges' across any of the tracks.

Firstly, using this single core wire, solder wire links between the points marked, one at a time. There are thirteen links in all. Since many of these links are close together, it is a good idea to use insulated wire where there is a possibility that they may touch.

IC sockets should be used for the three ICs as well as the interface sockets; there are seven in all. Insert and solder the IC sockets one at a time taking care not to allow the solder to 'bridge' across any tracks; special care needs to be taken when soldering the four in/out sockets.

The two IN4148 diodes (marked D1 and D2) should be soldered into position; these are tiny glass cased components. Note that one end of each is marked with a black band; this denotes the Cathode or negative lead; carefully check the PCB overlay to make sure that you mount these the right way round.

Resistor R1 should be inserted next. This is a larger, tubular component with coloured bands (Brown, Black, Red, Gold) and can be mounted either way round.

Solder in capacitors Cl and C2; these can be mounted either way round; they are larger than the resistor and, if they are not marked 0.1uf or 100nf, will be colour coded brown, black, yellow.

Finally, insert the edge connector (note the key position !!). This will take some patience since it is often difficult to line up the pins with their holes. Work from one end to the other, easing the pins into position one at a time.

Do not solder it in immediately but, using a ruler, make sure the lowest part of the connector body is spaced 7mm from the PCB surface. (See fig. 4). Make sure that the connec-

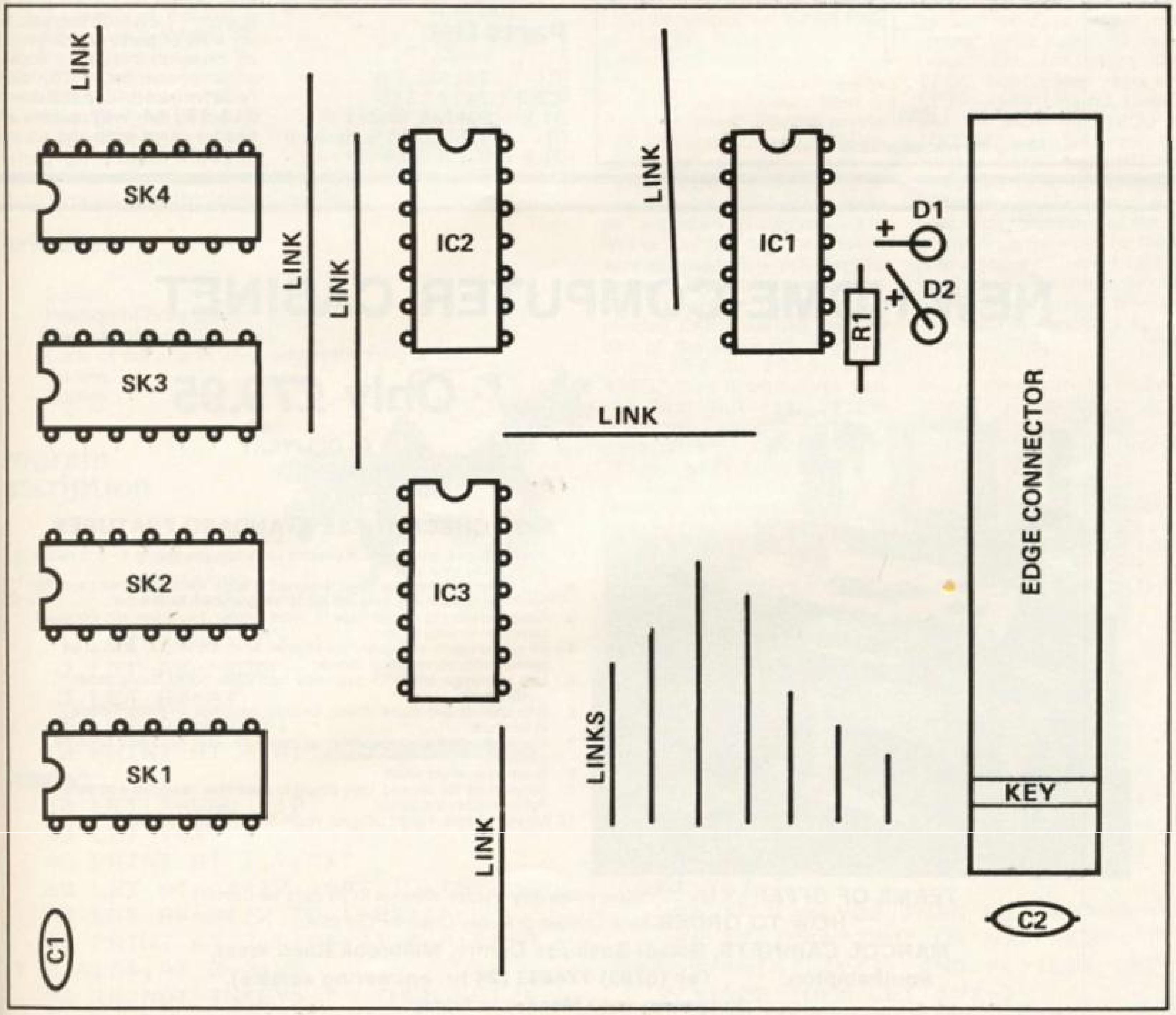


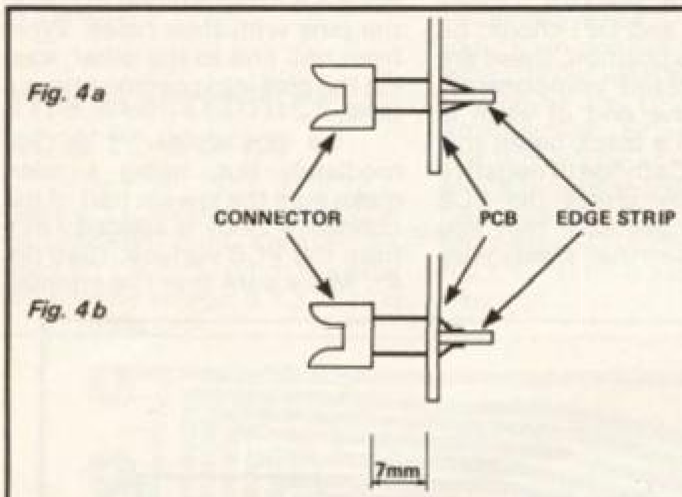
Fig. 3 Interface component overlay

tor is at right angles to the PCB and parallel with the PCB surface and then solder it in. Do NOT cut the connector pins if you wish to provide an edge connector for further peripherals to plug in behind your interface.

Now, settle down with a magnifying glass and a sharp fine tool (eg. craft knife), and spend half an hour checking that all your joints have been correctly soldered. Check for bridges between tracks and, if in doubt, use the knife to scratch between the tracks on the PCB. Be careful not to cut through any of the fine PCB tracks as you do this.

If you want to plug further peripherals in behind your interface (eg. ZX81 RAM pack) you should mount the rear edge connector strip (supplied in the kit) as follows:

Place this on the copper side of the PCB between the connector pins, lining up the pins of the connector with the tracks on the strip; make sure that the slot in the strip faces **away** from the PCB and is at the same end as the connector key.



Fitting the rear edge strip

With a pair of pliers or pincers, gently squeeze the two rows of pins together against the PCB strip so that the **ends** touch the copper (See fig. 4a). Now squeeze the pins again so that part of each pin lies flat on its corresponding track (See fig. 4b). Solder the strip to **all** connector pins taking care not to allow solder to flow back onto the PCB.

When you are happy that your interface has been constructed properly, you can plug in the three ICs. Each IC has a notch cut into one end of the case, (See fig. 3) this marks the top end with pin 1 at top left. Checking the PCB overlay, plug the ICs in one at a time in their correct positions. They will be marked 74LS32 or 74177 accordingly.

Be careful, when plugging the ICs in, that all pins go into the socket and that no pins become bent under the IC.

Parts List

IC1	74LS32 TTL
IC2,3	74177 TTL
D1,2	IN4148 Diodes
R1	1K 1/3W 5% Resistor
C1,2	0.1uf Capacitors

Miscellaneous

1 ZX81 type edge connector.

7 14 pin DIL IC sockets.

1 PCB.

(Up to four) Fourteen way jumper cables with DIP header plugs each end.

1 Rear edge connector strip, ZX81 type.

Buylines

All components used in this project with the exception of the PCB are readily available from electronic component suppliers who advertise regularly in electronics magazines.

The PCB can be obtained from Newtech Developments, Courtlands Road, Newton Abbot, Devon for £4.50 plus 30p postage. Newtech can also supply a kit of parts which includes all components, ICs, sockets, edge connector, PCB, diodes resistor and capacitors for £11.95, 14 way cables with header plugs each and are extra at £3.85 each.

The 1K Corral

Programmers of all ability levels can rope in ideas from this selection of the best of the least.

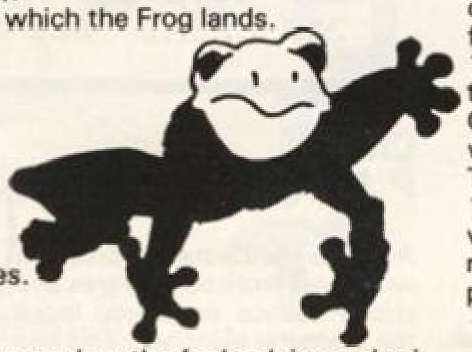


Frogger
Ian Arger

This is a version of the arcade game frogger which fits into 1K. When the program is run a river is printed with turtles swimming up and down. The frog which is shown as an "X" can be moved forward by pressing any key. The frog has to reach one of the homes on the other side of the river by jumping onto the backs of the turtles. If the frog jumps into the river, is washed down stream or misses the home, the game is over and the score is printed.

Variables

S	Score.
X	Position of Frog (Line).
Y	Position of Frog (Column).
P	Code of the character of which the Frog lands.
A$	Turtles.
B$	Turtles.



Program Description

1-5	Set up variables.
10-80	Prints river.
90-150	Moves Frog.
200-220	Adds to the score when the far bank is reached.

```
   1 LET S=NOT PI
   2 LET A$="OOO          OOO    "
   3 LET B$=A$
   5 LET A=NOT PI
  10 PRINT AT A,A;"▒▒▒▒█▒▒▒▒▒█▒▒▒▒▒█▒▒▒▒
▒▒█▒▒▒"
  30 LET X=VAL "10"
  40 LET Y=X
  45 PRINT AT X,Y;"X"
  60 LET A$=A$(17)+A$( TO 16)
  70 LET B$=B$(2 TO )+B$(1)
  80 PRINT AT 2,A;A$;AT 4,A;B$;A
T 6,A;A$;AT 8,A;B$
  90 IF NOT INKEY$="" THEN LET X
=X-2
 100 LET Y=Y+(X=2)+(X=6)-(X=4)-(
X=8)
 110 PRINT AT X,Y;
 120 LET P=PEEK (PEEK 16398+256*
PEEK 16399)
 130 PRINT "X"
 140 IF Y>16 OR Y<A OR P<9 OR (P
=61 AND X=A) THEN PRINT S;Q
 150 IF X=A THEN GOTO 200
 160 GOTO 50
 200 LET S=S+1
 210 IF S/4=INT (S/4) THEN GOTO
SQR PI
 220 GOTO 30
```

Day of the Week
P C Johnson

It's often useful to know which day of the week a particular date will fall on, but how often have you wondered on which day something happened in the past-most records only give dates. For example do you know on which day you were born — or your friends? The old saying tells us our fortune depends on which day we were born — "Wednesday's child is full of woe, Thursday's child has far to go" and so on. Would you like to find out which day the Bastille was stormed or which day the Battle of Waterloo was fought?

This program for the unexpanded ZX81 will tell you the day of the week for any date from1753 to 2099. Why 1753? Well because this was the first full year of the Gregorian Calendar — things were different before 1753! The program is listed in FIGURE 1. Note that in addition the variables listed in FIGURE 2 must be typed in as shown. The program must be used by typing GOTO 10 and never by RUN which would automatically CLEAR all the variables.

The program numbers the days from 1 for Sunday to 7 for Saturday and simply adds on the number of days above exact weeks for a particular month (VAL M$), for years since 1753, and for leap years since 1753 — all in Line 50. Lines 52 and 54 allow for 1800 and 1900 not being leap years (that is years in which Y/4 = INT(Y/4). Lines 56 and 58 look after dates in January and Febuary in a leap year. The remainder of the program brings the value for the day X to a figure of from 1 to 7 and converts this to the name of the corresponding day of the week.

Type in your date (line 10 and line 20) by:

Day of the month, NEWLINE,
Month — first three letters, NEWLINE.
Year — (all four figures), NEWLINE.

A second or two later your day of the week appears. It's a little like time travel. Good Hunting!

```
  10 PRINT "INPUT DATE:DAY,MONTH
,YEAR"
  20 INPUT D
  30 INPUT M$
  40 INPUT Y
  45 CLS
  47 PRINT "DATE: ";D;" ";M$;" "
;Y
  50 LET X=1+D+VAL M$+(Y-1753)+I
NT (Y/4)-438
  52 IF Y>=1800 AND Y<1900 THEN
LET X=X-1
  54 IF Y>=1900 THEN LET X=X-2
  56 IF (M$="JAN" OR M$="FEB") A
ND Y/4=INT (Y/4) AND Y<>1800 AND
 Y<>1900 THEN LET X=X-1
  60 IF X>7 THEN LET X=X-7
```

```
 70 IF X>7 THEN GOTO 60
 80 LET F$="IS"
 90 IF Y>=1985 THEN LET F$="WIL
L BE"
 92 IF Y<1984 THEN LET F$="WAS"

 95 PRINT
140 PRINT F$;" ";D$(X)+E$
```



Draw
Alan Ramsey

A short program which can amuse for ages, keys 5 and 8 draw left and right, keys 6 and 7 up and down. Pressing 2 allows you to add a keyboard character or characters to the drawing and key 9 will save the program (but not the drawing).

A great program to experiment with.

Variables

A. . Left and right position.
B. . Up and down position.
A$. . keyboard character(s).

Program Description

2-4 set variables.
3-30 cursor movement.
60 plot position.
70 go back to 5.
80-90 input characters.
100 go back to 5.

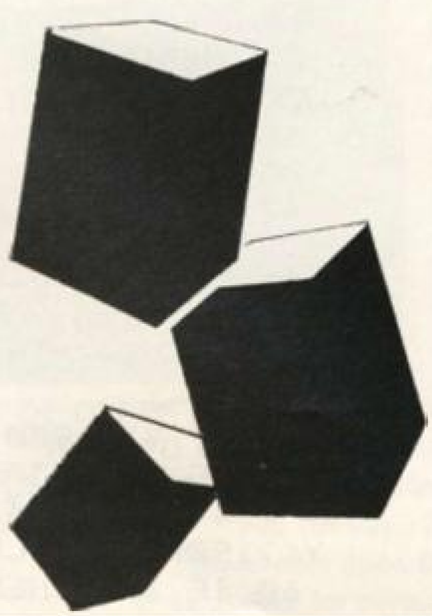
OK budding Picassos, let your artistic talents loose!

```
  2 LET A=0
  4 LET B=0
  5 LET A$=INKEY$
 10 LET A=A+(A$="8")-(A$="5")
 20 LET B$=INKEY$
 30 LET B=B+(B$="6")-(B$="7")
 40 IF INKEY$="2" THEN GOTO 80
 50 IF INKEY$="9" THEN SAVE "ZX
DRAW"
 60 PLOT A,B
 70 GOTO 5
 80 INPUT A$
 90 PRINT A$
100 GOTO 5
```

Bumper
Luuk Hilhorst

In this game it's your job to change the black blocks to "O" blocks. Running the game you will see some black blocks and a line. You move downwards, if you hit a black block it will change to an "O" block and you will come back to the upper side of the screen.

If you come below the line or if you hit an "O" block the game is over. For each block you change to an "O" block you get a point. If you changed 17 blocks to "O" blocks the screen will be cleared and other black blocks will appear.

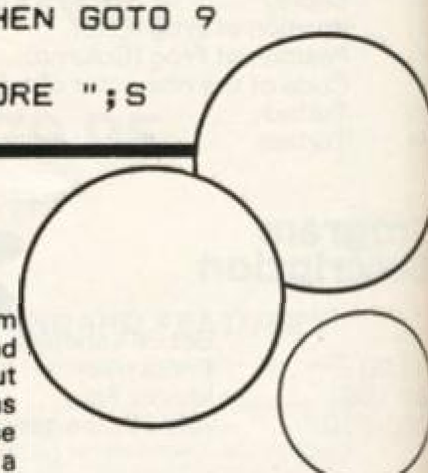


Program Description

Lines 10-40	Set up screen.
Line 70	Direction bump-plane.
Lines 80-90	What has the bump plane hit.
Lines 200-230	The routine if the bump-plane has hit a black block.
Line 250	Print score when the game is over.

```
  1 LET S=0
  9 CLS
 10 PRINT AT 14,0;"-----------
----"
 20 FOR F=1 TO 20
 30 PRINT AT RND*10+2,RND*15;"■
"
 40 NEXT F
 50 LET L=10
 60 FOR F=1 TO 14
 70 LET L=L+(INKEY$="8")-(INKEY
$="5" AND L>0)
 80 PRINT AT F,L;
 90 LET P=PEEK (256*PEEK 16399+
PEEK 16398)
100 PRINT "Y"
110 IF P=180 THEN GOTO 250
120 IF P=128 THEN GOTO 200
130 PRINT AT F,L;" "
140 NEXT F
150 GOTO 250
200 PRINT AT F,L;"▣"
210 LET S=S+1
220 IF S=17 THEN GOTO 9
230 GOTO 60
250 PRINT "SCORE ";S
```

Circle
P J Long



A simple mathematical program which will work out the area and circumference once you input the Diameter in inches. This has a few practical uses for those who want to know the area of a lawn or pool or patio which happens to be circular.

Line 3 takes in the diameter.
Line 6 multiplies and works out the formula 2*PI*r.

Line 9 applies the formula PI*(r*r) to get the area.

As an exercise you may like to alter the program to work in cm and metres.

```
  1 CLS
  2 PRINT AT 5,0;"INPUT DIAM. O
F CIRCLE IN INCHES"
  3 INPUT A
  4 CLS
  5 PRINT AT 5,0;"CIRCUMFERENCE
 OF CIRCLE IS"
  6 PRINT AT 8,0;(PI*2)*(A/2);"
 INCHES(";(PI*2)*(A/2)/12;" FEET
)"
```

```
   7 PRINT AT 15,0;"IF CIRCLE IS
 FLAT"
   8 PRINT AT 17,0;"THEN THE TOT
AL AREA OF CIRCLE IS";AT 18,15;"
■■■"
   9 PRINT AT 19,0;(PI*2)*(A/2)/
2*(A/2);" INCHES OR",,,(PI*2)*(A
/2)/2*(A/2)/12;" FEET"
  10 PRINT AT 0,5;"PRESS A TO RU
N"
  11 IF INKEY$="A" THEN RUN
  12 GOTO 11
  13 SAVE "CIRCLE"
  14 RUN
```

Target Practice
Dominic Carter



A variation on the reaction shoot 'em up games, type it in and RUN. As the target of numbers passes across the screen press 0 to fire at it. You score the number of the point you hit — 1,2 or 0.

```
  50 CLS
 100 LET L=INT (RND*25)+5
 200 LET B=19
 250 PRINT AT 20,L;"▄█▄"
 260 FOR T=1 TO 30
 261 IF INKEY$="0" THEN GOTO 400
 290 PRINT AT 0,T-1;"      "
 295 PRINT AT 0,T;"1 2 1"
 300 NEXT T
 350 GOTO 50
 400 FOR X=19 TO 0 STEP -1
 410 PRINT AT B,L;" ."
 420 IF B=0 AND L=T THEN GOTO 10
00
 430 IF B=0 AND L=T-2 THEN GOTO
1500
 440 IF B=0 AND L=T+2 THEN GOTO
1500
 500 LET B=B-1
 600 NEXT X
 700 GOTO 50
1000 CLS
1010 PRINT "BULLSEYE...2 POINTS"
1200 PAUSE 50
1300 GOTO 50
1500 CLS
1600 PRINT "YOU SCORED 1....NOT
BAD"
1700 PAUSE 50
1800 GOTO 50
```

Moon Rescue

Rescue the astronauts in this lunar listing for your Spectrum. Courtesy of John Miller.

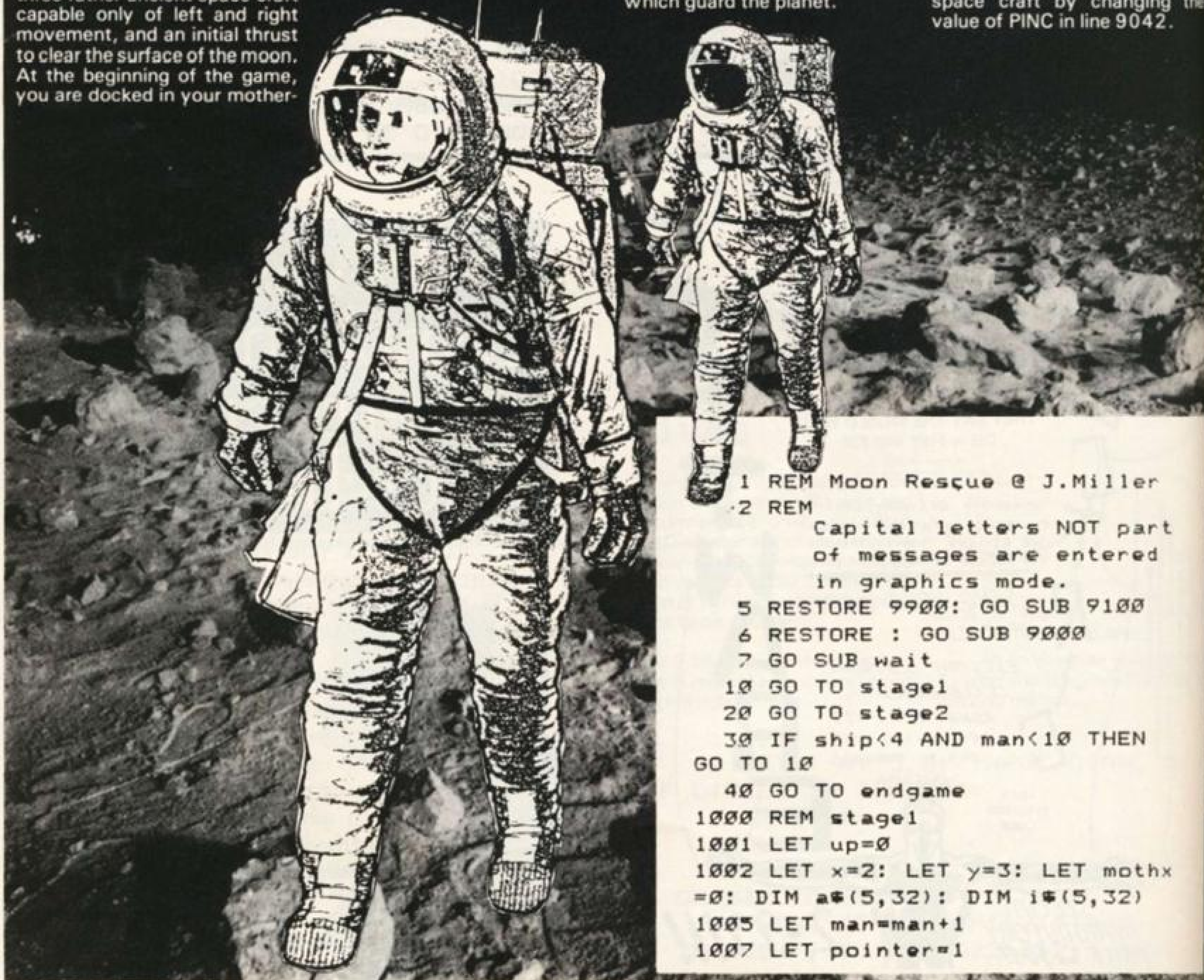


The object of this game is to rescue the 10 astronauts stranded on the moon. You have three rather ancient space craft capable only of left and right movement, and an initial thrust to clear the surface of the moon. At the beginning of the game, you are docked in your mothership traversing left and right at the top of the screen. Below are asteroid-like objects, and all you have to do is to get through this layer and you'll find the astronauts on the surface of the moon.

Moon madness

To start your descent, press the '0'; after that you can only move left and right using the '5' and '8' keys respectively — you descend automatically. During your descent you must avoid hitting the asteroids and then attempt to land your craft on the magenta landing pads. Once down safely, there will be a short delay while one of the astronauts clambers aboard. You then have to make your way back to the mothership, this time trying to avoid the cyan enemy ships which guard the planet.

To begin your ascent, press '0' key and then guide your ship using the cursor keys until you dock with the mothership. Once docked, you unload your human cargo and head off to rescue another astronaut. The game ends when you have either rescued all 10 astronauts or lost all three of your ships. Should you find the game all too easy, you could alter the velocities of the asteroids and the enemy space craft by changing the value of PINC in line 9042.



```
   1 REM Moon Rescue @ J.Miller
   2 REM
       Capital letters NOT part
       of messages are entered
       in graphics mode.
   5 RESTORE 9900: GO SUB 9100
   6 RESTORE : GO SUB 9000
   7 GO SUB wait
  10 GO TO stage1
  20 GO TO stage2
  30 IF ship<4 AND man<10 THEN
GO TO 10
  40 GO TO endgame
1000 REM stage1
1001 LET up=0
1002 LET x=2: LET y=3: LET mothx
=0: DIM a$(5,32): DIM i$(5,32)
1005 LET man=man+1
1007 LET pointer=1
```

```
1ØØ8 LET inc=1
1Ø1Ø FOR c=1 TO 5: FOR d=1 TO 32
1Ø2Ø IF  RND *(11-man)<.4 THEN
LET a$(c,d)="G"
1Ø3Ø NEXT d: NEXT c
1Ø35 CLS
1Ø4Ø GO SUB surface
1Ø5Ø GO SUB topprint
1Ø6Ø PRINT  AT Ø,6; INK 6;score;
 AT Ø,17; INK 5;high; AT Ø,3Ø; I
NK 7;ship
1Ø7Ø FOR c=1 TO 5: PRINT  AT 4+c
*2,Ø; INK 6;a$(c): NEXT c
1Ø8Ø PRINT  AT 2,Ø;m$; AT 3,Ø;n$
; AT y,x
1Ø85 PRINT  AT 3,3;s$
12ØØ LET mothx=mothx+inc: IF  NO
T drop THEN  LET x=x+inc
12Ø1 IF mothx=28 THEN  LET inc=-
1
12Ø2 IF mothx=1 THEN  LET inc=1
12Ø5 PRINT  AT 2,mothx;m$; AT 3,
mothx;n$: PRINT  AT y,x;s$
121Ø LET pointer=pointer+pinc: I
F pointer>32 THEN  LET pointer=1
1215 FOR c=1 TO 5: PRINT  INK 6;
 AT 4+c*2,Ø;a$(c,pointer TO );a$
(c, TO pointer-1): NEXT c
122Ø IF  INKEY$ ="Ø" THEN  LET d
rop=1
1221 BEEP .ØØ5,24*drop-12: BEEP
.ØØ5,24*drop-1Ø
1223 PRINT  AT y,x;" "
1225 LET y=y+drop: IF y=21 THEN
 GO TO shiploss
1227 IF  NOT drop THEN  GO TO 12
ØØ
123Ø LET x=x-( INKEY$ ="5" AND x
>Ø)+( INKEY$ ="8" AND x<32)
1235 IF  ATTR (y,x)=3 THEN  GO T
O 2Ø
124Ø IF  SCREEN$ (y,x)="" THEN
GO TO shiploss
13ØØ GO TO 12ØØ
1999 STOP
2ØØØ PRINT  AT y-1,x;s$: FOR c=1
 TO 1Ø: BEEP .1,-1Ø+c: BEEP .1,-
1Ø-c: NEXT c
2ØØ1 LET up=Ø: LET drop=Ø
2ØØ2 LET pointer=1
2ØØ5 LET m(nman)=Ø: LET nman=nma
n+1
2Ø1Ø CLS : GO SUB surface: GO SU
B topprint
2Ø2Ø LET y=y-1: PRINT  AT y,x;s$
2Ø3Ø FOR c=1 TO 5: FOR d=1 TO 32
2Ø4Ø IF  RND *(11-man)<.4 THEN
LET i$(c,d)="H"
2Ø5Ø BEEP .ØØ5,3Ø: NEXT d: NEXT
c
2Ø55 LET score=score+25Ø
2Ø6Ø PRINT  AT Ø,6; INK 6;score;
 AT Ø,17; INK 5;high; AT Ø,3Ø; I
NK 7;ship
2Ø7Ø FOR c=1 TO 5: PRINT  AT 4+c
*2,Ø; INK 5;i$(c): NEXT c
2Ø8Ø PRINT  AT 2,mothx;m$; AT 3,
mothx;n$
21ØØ LET mothx=mothx+inc
21Ø1 IF mothx=28 THEN  LET inc=-
1
21Ø2 IF mothx=1 THEN  LET inc=1
21Ø5 PRINT  AT 2,mothx;m$; AT 3,
mothx;n$
21Ø6 PRINT  AT y,x;s$
211Ø LET pointer=pointer+pinc: I
F pointer>32 THEN  LET pointer=1
2115 FOR c=1 TO 5: PRINT  INK 5;
 AT 4+c*2,Ø;i$(c,pointer TO );i$
(c, TO pointer-1): NEXT c
212Ø IF  INKEY$ ="Ø" THEN  LET u
p=1
213Ø BEEP .ØØ5,24*up-12: BEEP .Ø
Ø5,24*up-1Ø
214Ø PRINT  AT y,x;" "
215Ø LET y=y-up: IF y<3 THEN  GO
 TO shiploss
216Ø IF y=3 AND x=mothx+2 THEN
GO TO home
217Ø IF  NOT up THEN  GO TO 21ØØ
218Ø LET x=x-( INKEY$ ="5" AND x
>Ø)+( INKEY$ ="8" AND x<32)
219Ø IF  SCREEN$ (y,x)="" THEN
GO TO shiploss
22ØØ GO TO 21ØØ
2999 STOP
3ØØØ INK 7: PAPER Ø: BORDER Ø: C
LS
3Ø1Ø FOR c=Ø TO 5Ø: PLOT  RND *2
55, RND *175: NEXT c
3Ø2Ø PRINT  AT 5,4; INK 5;"WELCO
ME TO 'MOON RESCUE'"
3Ø25 PRINT  AT 7,4; INK 3;"@ Joh
n Miller  Oct 1982"
3Ø3Ø PRINT  AT 16,Ø; FLASH 1; BR
IGHT 1;"PRESS ANY KEY WHEN YOU A
RE READY"
3Ø4Ø IF  INKEY$ ="" THEN  GO TO
3Ø4Ø
3Ø45 PRINT  AT 16,Ø; OVER 1;"PRE
SS ANY KEY WHEN YOU ARE READY"
3Ø5Ø RETURN
4ØØØ INK 4: PRINT  AT 17,Ø;"I
                              M"
```

```
4ØØ1 PRINT  AT 18,Ø;"■I
                   M■"
4ØØ2 PRINT  AT 19,Ø;"■■I
                  M■■"
4ØØ3 PRINT  AT 2Ø,Ø;"■■■I    "; I
NK 3;"K■■N"; INK 4;"         "; I
NK 3;"K■■N"; INK 4;"     M■■■"
4ØØ4 PRINT  AT 21,Ø;"■■■■JJJJ■■J
JJJJJJJJJ■■JJJJJJ■■■■"
4ØØ5 FOR c=1 TO 1Ø: IF m(c)>Ø TH
EN  PRINT  AT 2Ø,m(c); INK 6;"L"
4ØØ6 NEXT c
4Ø1Ø INK 7: RETURN
5ØØØ PRINT  AT Ø,Ø; INK 6; BRIGH
T 1;"SCORE:        "; INK 5;"HIGH:
     "; INK 7;"SHIP:  "
5Ø1Ø PRINT  INK 3;"
                    "
5Ø2Ø RETURN
55ØØ FOR c=Ø TO 7 STEP .5: PRINT
  AT y,x; INK c;"E": BEEP .Ø1,c*
5: BEEP .Ø1,c*-5: NEXT c
551Ø LET score=score+3ØØ: GO TO
3Ø
6ØØØ FOR c=y TO 2Ø: PRINT  AT c,
x; INK  RND *7;"E"; OVER 1; AT c
,x;"E": BEEP .Ø1,-c: NEXT c
6ØØ5 BEEP 1,-2Ø: PAUSE 5Ø
6Ø1Ø LET ship=ship+1: LET drop=Ø
: LET up=Ø: CLS : GO TO 3Ø
8ØØØ PRINT  AT 6,11; FLASH 1;"GA
ME OVER"
8Ø1Ø IF score>high THEN  LET hig
h=score: FOR c=1 TO 1Ø: BEEP .Ø1
, RND *5Ø: BORDER  RND *7: NEXT
c: BORDER Ø: PRINT  AT 11,Ø; BRI
GHT 1; FLASH 1;"WELL DONE - A NE
W HIGH SCORE!!!!"
8Ø2Ø PAUSE 25Ø: CLS : GO TO 5
8999 STOP
9ØØØ DIM a$(5,32): DIM i$(5,32)
9Ø1Ø LET ship=1: LET man=Ø: LET
x=2: LET y=3: LET nman=1
9Ø15 LET s$= CHR$ 19+ CHR$ 1+"E"
: LET m$=" "+ CHR$ 16+ CHR$ 2+"A
BC ": LET n$=" "+ CHR$ 16+ CHR$
2+"D F "
9Ø2Ø LET mothx=Ø: LET drop=Ø: LE
T score=Ø
9Ø3Ø LET stage1=1ØØØ: LET stage2
=2ØØØ: LET endgame=8ØØØ
9Ø4Ø LET wait=3ØØØ
9Ø42 LET pinc=2
9Ø45 LET surface=4ØØØ
9Ø47 LET topprint=5ØØØ: LET ship
loss=6ØØØ
9Ø48 LET home=55ØØ
9Ø5Ø DIM m(1Ø): FOR c=1 TO 1Ø
9Ø6Ø READ m(c): NEXT c
9Ø7Ø DATA 4,5,6,11,14,15,17,23,2
5,26
9Ø8Ø RETURN
91ØØ FOR c= USR "a" TO  USR "n"+
7: READ a: POKE c,a: NEXT c
91Ø1 LET high=Ø
9899 RETURN
99ØØ DATA  BIN Ø, BIN 1ØØ, BIN Ø
Ø111111, BIN 1111111, BIN 11ØØØØ
Ø, BIN 111Ø11Ø, BIN 1111111, B
IN 1111111
99Ø1 DATA  BIN 11ØØØ, BIN 11111
Ø, BIN 1111111, BIN 1111111, B
IN Ø, BIN 1111111, BIN 11ØØ111
, BIN 11ØØØ11
99Ø2 DATA  BIN Ø, BIN 1ØØØØØ, BI
N 11111ØØ, BIN 111111Ø, BIN 11
1, BIN 11Ø1111, BIN 1111111, BI
N 1111111
99Ø3 DATA  BIN 11111ØØØ, BIN 111
11ØØØ, BIN 11111ØØØ, BIN 11111ØØ
, BIN 111111, BIN 11111, BIN 111
1, BIN 111
99Ø4 DATA  BIN 1ØØØ, BIN 111ØØ,
BIN 1111111, BIN 1ØØ1ØØ1, BIN 1Ø
111Ø1, BIN 11111Ø, BIN 1ØØØ1Ø, B
IN 1ØØØØØ1
99Ø5 DATA  BIN 11111, BIN 11111,
 BIN 11111, BIN 11111Ø, BIN 1111
11ØØ, BIN 11111ØØØ, BIN 1111ØØØØ
, BIN 111ØØØØØ
99Ø6 DATA  BIN 11ØØØ, BIN 111111Ø
Ø, BIN 1111111Ø, BIN 1111111, BI
N 1111111, BIN 111111Ø, BIN 111Ø
ØØ, BIN 1ØØØØ
99Ø7 DATA  BIN Ø, BIN 1111ØØ, BI
N 11ØØ11Ø, BIN 1111111, BIN 111
11111, BIN 1Ø1111Ø1, BIN 1ØØØØ1Ø
, BIN 1ØØ1ØØ
99Ø8 DATA  BIN 1ØØØØØØØ, BIN 11Ø
ØØØØØ, BIN 111ØØØØØ, BIN 1111ØØØ
Ø, BIN 11111ØØØ, BIN 11111ØØØ, B
IN 1111111Ø, BIN 11111111
99Ø9 DATA  BIN 1, BIN 1Ø11ØØ11,
BIN 11111111,255,255,255,255,255
991Ø DATA  BIN 111111, BIN 1111
11, BIN 11111, BIN 1111, BIN 111
, BIN 111, BIN 1, BIN Ø
9911 DATA  BIN Ø, BIN 111ØØ, BIN
 111ØØ, BIN 1ØØ1ØØ1, BIN 11111Ø,
 BIN 1ØØØ, BIN 1Ø1ØØ, BIN 1ØØØ1Ø
9912 DATA  BIN 1, BIN 11, BIN 11
, BIN 1111, BIN 11111, BIN 11111
1, BIN 1111111, BIN 1111111
9913 DATA  BIN 1111111Ø, BIN 111
111ØØ, BIN 11111ØØ, BIN 1111ØØØ
Ø, BIN 111ØØØØØ, BIN 11ØØØØØØ, B
IN 1ØØØØØØØ,Ø
9999 RETURN
```

Renum

Many renumber programs have been published but this one has an unusual feature — read Geoffrey Byrne's explanation.



I have discovered something I have not read about elsewhere. The Sinclair manual says that the highest line number allowable is 9999. Although this is so, in practice lines higher than this may be utilised using the Renumber Routine which follows.

An advantage of this is that the lines pushed over the 10000 limit cannot be re-edited and are therefore safe from tampering. This could be useful in preserving data held in strings etc.'

Enter the following program:

```
10  REM (47 x's)
20  REM ***LOADER***
30  FOR J=16516 TO
    16560
40  INPUT N
50  PRINT AT 18,8;J,N
60  SCROLL
70  POKE J,N
80  NEXT J
FAST
```

Enter the following data checking carefully as you go:

```
33, 125, 64, 17, 0, 0, 126,
230, 192, 192, 229, 70,
35, 78, 35, 235, 229, 237,
66, 225, 56, 18, 193, 124,
2, 3, 125, 2, 1, 10, 0, 9,
235, 78, 35, 70, 35, 9, 24,
222, 96, 105, 193, 24,
239.
```

Now enter, as direct commands:

```
POKE16514,0
POKE16515,0
SAVE "RENUMBER"
```

Having SAVEd the programme, enter RUN and then the following:

```
9990 REM
9991 FOR J = 1 to 15
9992 PRINT "ZX81 RULES
O.K."
9993 NEXT J
```

Check this by entering:

```
SLOW
RUN 9990
```

Now enter RAND USR 16516 and LIST.

You should now find some very odd looking numbers in the listings. The listing should be as follows:

```
A000 REM
A010 FOR J = 1 TO 15
A020 PRINT "ZX81 RULES
O.K."
A030 NEXT J
```

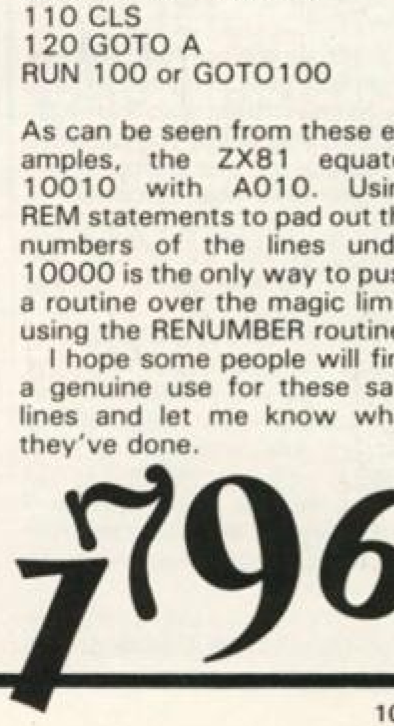
As you can see, the '10' of 10000 has been replaced by 'A' as in Hexadecimal, but the other numbers remain in the decimal! Since the first character of the line number is not recognised by the ZX81 as a number, it cannot be accessed for editing, and is therefore incorruptable once placed over 10000.

These lines can be accessed from within a program. Enter the following:

```
100 LET A = 10010
110 CLS
120 GOTO A
RUN 100 or GOTO100
```

As can be seen from these examples, the ZX81 equates 10010 with A010. Using REM statements to pad out the numbers of the lines under 10000 is the only way to push a routine over the magic limit, using the RENUMBER routine.

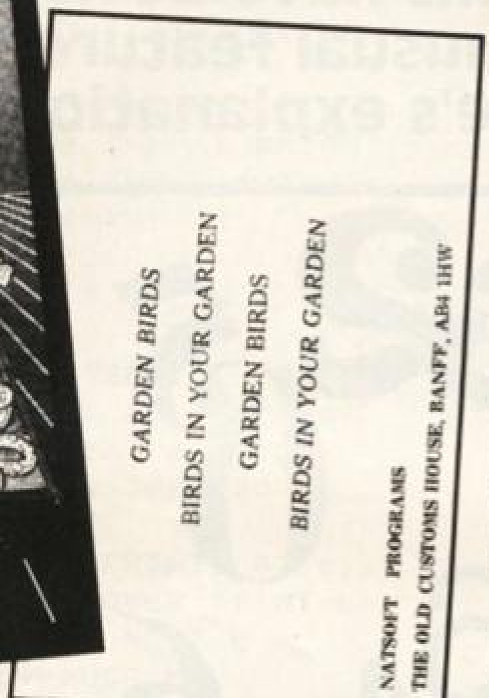
I hope some people will find a genuine use for these safe lines and let me know what they've done.

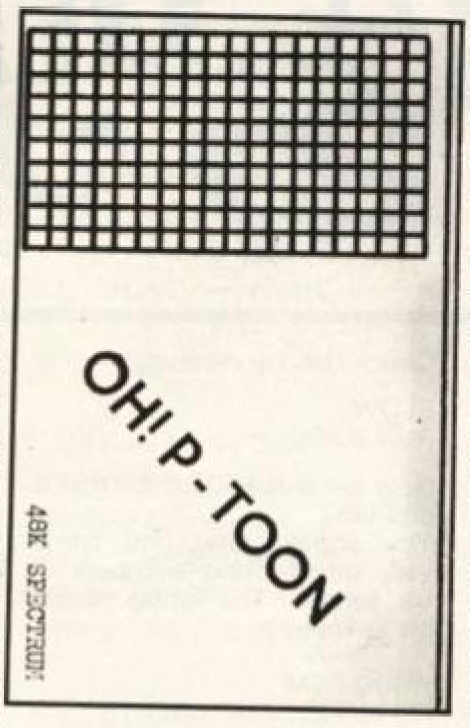


Quicksoft

Rapid reports, courtesy of Clive Smith







Toolkit Kerian U.K. Ltd.

For those of you who do a lot of programming this is a must. If you have never ever seen a tool kit program I'll give you the run down.

If you are about to write a program you first load in the tool kit program. A list of ten options appear on screen. One is auto line numbering which automatically prints your next line number in any increment you wish.

Another is renumber, this will renumber your program after it has been written, also in any increment you wish. You can also put a trace on, this prints the line number that the computer is reading when your program is running at the bottom of the screen. This is a must for finding bugs or errors.

Took kit also has a couple of search routines, one will find any strings you want to find in your program, and another will find any key words ie: GOTO, POKE.

One of the best options I found was a variable list, this lists all your variables and gives their values.

There are other options which will delete blocks of line numbers, tell you how much memory you have left and a delete option which only knocks out all your REM lines.

All in all a very useful cassette to have for the avid programmer. It can only be used on the 48K Spectrum and will ser you back £4.95.

If you don't happen to have a 48K Spectrum there are many books which have some of the options listed here which you can enter yourself.

Garden Birds Natsoft Programs

It ceases to amaze me the applications the home computer has been put to. Now for all prospective ornithologists you can get "Garden Birds", a bird spotting programme.

The tape is to be used in conjunction with the RSGB bird book. The idea is to go out in the garden and throw some breadcrumbs on the path, then retire to the lounge window and wait for the birds to arrive. When you see a bird that you can't identify you then turn to the computer. It asks you if the bird you spotted is the same size as:

A. A Sparrow
B. A Blackbird or
C. A Pigeon.

Once you have established one of these options and have pressed the appropriate key the screen clears and you are given several more options. This applies to the colour of the bird, let's say our bird is black and white and we press the key for that colour. You get a third set of options asking for more details such as beak shape etc. Once all the details havew been entered it then makes a guess at the bird you had seen.

Now, being a bit of a bird spotter myself, I gave it a try and gave all the details for a bullfinch and it told me I had seen a Chaffinch. You see it doesn't hold a Bullfinch in it's memory. In fact it only holds about 60 birds in memory, which is not a great many considering the number of species there are. When it does tell you the bird you have seen, it tells you which page you will find it in the RSGB Bird book. I think you would be better off spending your money on a RSGB Bird book which lists all the birds with illustrations.

Music Maker Malan PO Box 390 Purleigh. ESSEX

There are many music making devices about, mainly in the hardware department but you don't see too many on the software side. Well here's one, produced by Malan.

Written for the 48K machine and with the novice composer in mind, Music Maker is a quick and simple way of making quick and simple tunes.

If you're hoping to write the equivalent of Beethovens 5th then forget it. It is very limited with one channel sound and a maximum of about 60 notes (use any more and it crashes). The top half of the screen displays a blank song sheet to put the notes on. On the lower half is printed your options either to play, move backwards, forwards, on save and load your mini tune. Notes can be made to sound longer or shorter and it has the ability to produce a 'pip' and 'hum'. If composing is your scene, save some money and buy something better. But if you want something that with a little patience you could produce with beep statements, then maybe this will suffice.

OH! P—Toon Leon-Noel 24 Dudgeon Drive, Littlemore Oxford

Ever seen one of those pontoon games printed in computer books? Well, this is a posh version of one of those.

Not many points for originality but I did like the graphics. What they have done is taken the basic game of pontoon and given it a new twist. *(Keep it up, Clive — Ed.)*

What you have to do is watch three cards which appear one after another and add up the total. If the cards are less than 21 you have to press the key B, if it adds up to 21 you have to press 'N and if you have over 21 you have to press M.

Points are awarded for speed in adding up the cards, the faster you can add up the cards the more points you get. *(Sounds exciting doesn't it?)*.

If you get bored you can change the rate of points you get. Never mind, the graphics are great. After you have loaded the tape, you have the option of playing the game or reading the instructions. One small bug here. When you have read the instructions you have to press a key to move onto the game, but when you do this it crashes! OH! P-TOON is written for the 48K Spectrum and will set you back £4.95.

UNIQUE Competition

Anyone and Everyone Stands to Win!!

Yes, UNIQUE the latest and most original company to launch a range of programs are not only giving away £600 worth of their latest program but everyone who enters gets free membership to the "Thats Different" club which entitles you to a regular monthly round-up of "Whats New" and of course reduced prices on the complete range of UNIQUE products!!

UNIQUE'S range of programs consist of Clerky, Sand-Scorchers, Whizz-Kid and Bully all for the 48K Spectrum except Whizz-Kid which is also 16K and, as you can gather from the name, are quite UNIQUE!!

"Most of our programs", says UNIQUES General Manager, "are written by freelance programmers happily working under contract but we also have a team of in-house programmers working full time as well as graphic design artists and a marketing analysis team who all work together to produce a range of programs that are of high quality and represent good value for money".

So don't delay enter today and win **your** Sand-scorcher game worth £5.99.

Problem

There are 5 names of top software houses and 5 names of top programs. All you have to do is to find the names and games and link them together.

It's so simple!!!

Answers on a postcard please, addressed to:

ZX Computing
Unique Competition,
1, Golden Square,
London W1R 3AB.

The Rules

- This competition is open to all UK and Northern Ireland readers of *ZX Computing*, except employees of Argus Specialist Publications Ltd. Their printers and distributors, employees of Unique or anyone associated with the competition.
- As long as the correct coupon is used (a postcard in this case), there is no limit to the number of entries from each individual.
- All entries must be postmarked before September 30th 1984. The prizes will be awarded to the first hundred entries picked at random which bear the correct answers, the decision to be made by the Editor of *ZX Computing*. No correspondence will be entered into with regard to the results and it is a condition of entry that the Editor's decision is accepted as final.
- The winners will be notified by post and the results will be published in a future issue of *ZX Computing*.

Competition Results

The 'write a pun' competition (*ZX Computing* April/May) had wordsmiths up and down the country conjuring computer-couplets, so a big thank you to all ZX' readers who posted their puns to us. As you may recall, the prize on offer was a superb Marcol computer cabinet, and one of these will be on its way to **Mr. G.J. Evans** of Prestbury, Cheltenham, who managed to fit a staggering 25 puns on his postcard. We educated folk are not impressed by sheer numbers of course, but Mr. Evans' entries were by far the best of the bunch. Take for example, his 'Beatles Computer Album' which, amongst other tracks includes:

Let it BEEP
SCROLL OVER Beethoven
YELLOW Subroutine
YesterDATA
I Wanna Hold Your RAND
and
LN OR Rigby.

Also, Mr. Evans illustrates his taste for the classics by mentioning Mussorgsky's 'Pixels at an Exhibition', and he also expresses a liking for a little easy-listening with Frank Sinatra's 'Compile With Me'. (I know who *won't* be Betjemans' successor as poet laureate — ED).

Close-seconds were: **Mrs. J. Stoddart** of Edinburgh, for her variation on the Kung-Fu classic 'Enter the Jargon' (quite appropriate!); **Carol Rice** of Birmingham, who also sent us a healthy selection of quips, suggests 'Screen Dumps of Lily' by The Who — a personal favourite — and 'The hole in my CPU was letting in DATA'; **Mr. N. Robertson** of Stirlingshire, Scotland, and **G. Norton** of Norwich both chose Rock group Queen as subjects for a play on words with 'Another One BYTES the Dust'. Two other entrants chose The Hobbit as a key phrase; **R.H. Young** of Surrey suggested 'Hobbit, Hobbit, Hobbit' by Chas and Dave, while **S. Salehmohamed** of Bolton chose 'Run Hobbit, Run Hobbit, Run, Run, Run' by (wait for it...) BUGs Bunny.

Also from **R.H. Young** (you could have used the same postcard!) came 'EAR on a G$', while 'Sinclair on a G$' from **Bill Clarke** of Nottingham suggested a similar idea.

Lastly, there were a number of entries which were, to put it politely, obscure. I'm DIM IN OVER WHITE CHR$ MERGE FROM **Colin Nesbit** was one of few which were almost completely composed of ZX keywords (an OLD BINCHR$ PI number, incidently!). Now, an entry from **John Stuart** of Ayr was baffling in the extreme. John asks 'what about that famous modern musical:

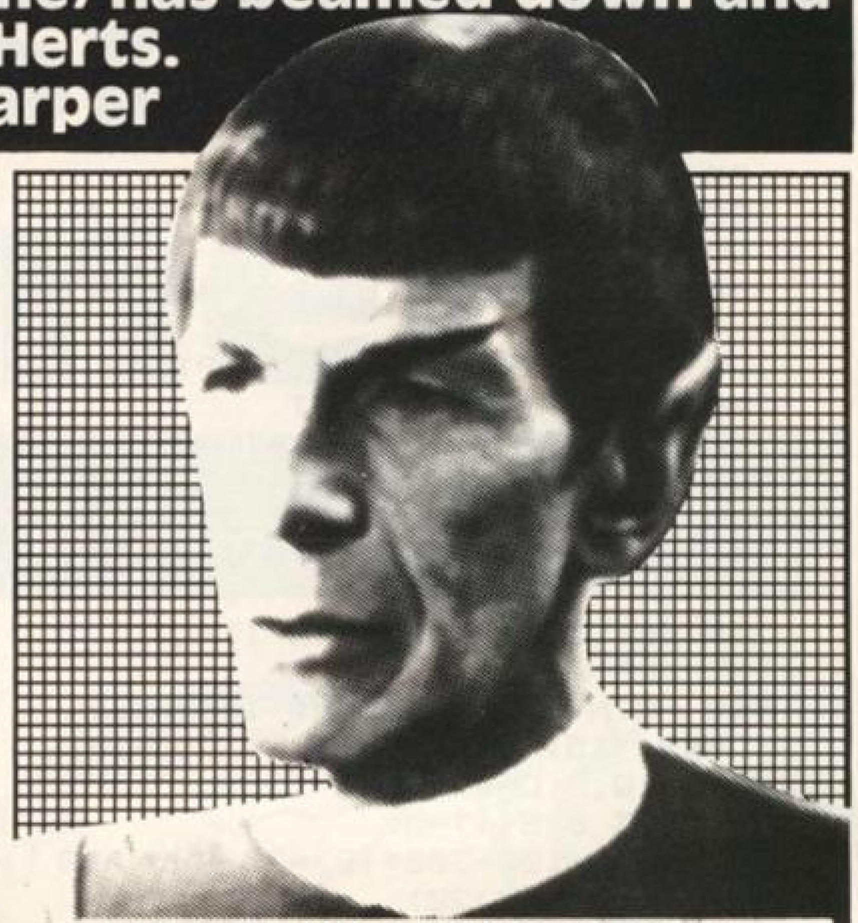
GOSUB AND THEN ARRAY STEP INK SCROLL OR dREM CODE. — DIM PIce & RAND NEW LOAD RED OR.

?????? What can I say? Any reader who can suggest the true meaning *might* receive a special prize from us!

Thanks again to all those who entered the competition. A special word of thanks goes to **Mr. D.E. Powis** of West Norwood in London for his poem 'I had a Dream'. Space does not permit it's publication, but a fine poem all the same.

U	L	T	I	M	A	T	E	R	O	L	F	Q	W	D	Y
E	L	O	M	S	O	V	M	E	S	U	I	P	Q	Z	R
O	C	E	A	N	P	A	P	N	O	V	O	G	A	M	E
N	U	O	G	O	P	A	U	I	A	N	Z	X	C	B	K
V	R	B	I	Z	O	T	M	M	M	O	J	R	T	I	V
P	T	O	N	Y	S	R	O	C	C	E	G	R	Y	R	O
A	A	U	E	B	R	V	M	I	V	W	A	C	R	S	B
X	W	W	Z	C	I	O	R	N	S	A	A	B	V	A	Z
O	R	S	B	S	M	P	R	A	S	T	B	B	T	X	W
A	I	D	A	C	R	A	P	M	O	U	M	I	P	R	U
Z	Y	D	O	U	W	T	Q	W	V	J	E	T	P	A	C
E	C	R	L	V	S	R	R	F	V	W	A	B	Z	I	S
F	A	G	H	O	Y	U	R	V	X	S	E	R	U	T	B
P	G	Q	R	Z	X	C	Z	X	G	I	R	V	J	L	N
S	V	M	H	D	F	K	F	T	L	K	C	M	W	H	Q
S	T	C	E	J	O	R	P	E	R	A	W	T	F	O	S

Mr. Spock (of Star Trek fame) has beamed down and taken up residence in Herts. under the name of A J Harper



An example of a logical expression is shown in line 30 of figure 1. What does it do? If in doubt either read page 68 of the ZX81 manual or run lines 10 to 50 of figure 1 (Type RUN 10, and enter any two numbers for A and B). Running lines 10 to 50 demonstrate that if the value of A is greater than the value of B then L will be set to the value one. However if the value of A is less than or equal to the value of B then L will be set to zero.

By using the operators AND, OR, NOT, complex logical expressions may be developed. Logical expressions may be put to many uses some of which are illustrated in the following examples.

1 : **A Computed GOTO** : (See lines 100 to 145 of figure 1). To run this example enter RUN 100 and enter a value for A. If A is set to 3 then line 120 will cause the programme to jump to line 130. If A is set to 4 then line 120 will cause a jump to line 140. Any other value for A will cause the programme to jump to line 110.

2 : **PRINT control** : (See lines 200 to 230 of figure 1). To run this example enter RUN 20 and enter a value for A. Line 120 will cause either the inverse video of character 27 or character 27 to be printed depending on the value entered for A.

3 : **Movement control** : (See lines 300 to 330 of figure 1). To run enter RUN 300. The PLOT character can then be controlled using cursor keys 5 and 8. The code in line 310 will cause the extent of movement to be constrained between horizontal values of zero and thirty.

4 : **Function Generation** : (See lines 400 to 440 of figure 1).
Suppose it is necessary to construct a function such that,

Y = 10 + X, for values of X less than 10
Y = 20, for values of X between 10 and 20
Y = 40 − X, for values of X greater than 20

This task is accomplished by the line of code in line 420 of figure 1. Enter RUN 400 and a graph of Y against X will be plotted.

5 : **An "up market" space invaders in a 1K ZX81** : (See figure 2).
These ideas and other examples have been combined together in figure 2 in a space invaders type game for a 1K ZX81. The missile base is moved using keys 5 and 8. The missile fire button is key 0. So what's new? Most invaders in 1K games are unarmed. In this game, the invaders fire back with guid-

Fig. 1 Program containing examples of use of logical expressions

```
 5 REM A LOGICAL EXPESSION
10 INPUT A
20 INPUT B
30 LET L=(A>B)
40 PRINT L
50 STOP
99 REM
100 REM EG1:A COMPUTED GOTO
110 INPUT A
120 GOTO 110+20*(A=3)+30*(A=4)
130 PRINT "LINE 130",A
135 STOP
140 PRINT "LINE 140",A
145 STOP
199 REM
200 REM EG2:PRINT CONTROL
210 INPUT A
220 PRINT CHR$ (27+128*(A=0))
230 STOP
299 REM
300 REM EG3:MOVEMENT CONTROL
305 LET X=15
310 LET X=X+(INKEY$="8")*(X<=30
)-(INKEY$="5")*(X>=0)
320 PLOT X,10
325 UNPLOT X,10
330 GOTO 310
399 REM
400 REM EG4:FUNCTION GENERAION
410 FOR X=1 TO 40
420 LET Y=(10+X)*(X<10)+20*(X>=
10 AND X<=20)+(40-X)*(X>20)
430 PLOT X,Y
440 NEXT X
```

Logical Expressions

```
  10 LET X=0
  20 LET Y=0
  30 LET A=X
  40 LET B=Y
  50 LET P=10
  60 LET Q=21
  70 LET S=0
 100 LET L=INKEY$="0" OR Q<21 AN
D Q>Y
 110 LET C=RND<.1 OR B>Y AND B<2
0
 120 LET T=Y<Q OR ABS (X-P)>=2
 200 LET X=(X+1-RND)*T
 210 LET Y=(Y+.25+RND)*T
 220 LET A=X*(1-C)+(A+(P-A)/5)*C
 230 LET B=Y*(1-C)+(B+1.25)*C
 240 LET P=P+(INKEY$="8")-(INKEY
$="5")
 250 LET Q=21*(1-L)+(Q-1)*L
 300 CLS
 310 PRINT AT Y,X;CHR$ 151;AT B,
A;CHR$ (151-128*C);AT 21,P;CHR$
149;AT Q,P;CHR$ (149-128*L)
 320 LET S=S+(T=0)
 330 GOTO 100+300*(B>=20 AND ABS
 (A-P)<2 OR Y>20)
 400 CLS
 410 PRINT S
```

Fig. 2 Space Invaders illustrating use of logical expressions

ed missiles. Just try leaving the missile base still and watch the asterisk launched by the invader converge onto your base position. How's it done? The secret is contained in line 140. When in flight the horizontal position of the invaders missile is obtained from the expression:

A = A + (P − A)/5

In this expression P is the horizontal position of the base and A is the horizontal position of the missile. Clearly if A = P then A will remain unchanged and the missile will strike the base. Now work out what happens if the missile is to the right or left of the base. Can you escape the missile by moving the base? Try it and see.

Finally there are three expressions which can be of some use when trying to either reduce the complexity of a logical expression or change the value of a logical expression from zero to one. They are shown in figure 3.

For example, in the original version of example 5, lines 120 and 200 read as shown in figure 4. Here Z = 0 when the invader is in flight. However a more economical coding is shown in line 200 of figure 2. In other words Z should be changed to T, where T = 1 when the invader is in flight.i.e. T = NOT(Z). This can be done by thinking (which can be difficult!) or by using expression (b) of figure 3. (The working is shown in figure 5).

(X AND Y) OR (X AND Z) = X AND (Y OR Z)
NOT(X AND Y) = NOT(X) OR NOT(Y)
NOT(X OR Y) = NOT(X) AND NOT(Y)

Fig. 3 Logical Identities

```
120   LET Z=(Y>=Q) AND ABS(X-P)<2
200   LET X= X+1-RND) *(1-Z)
```

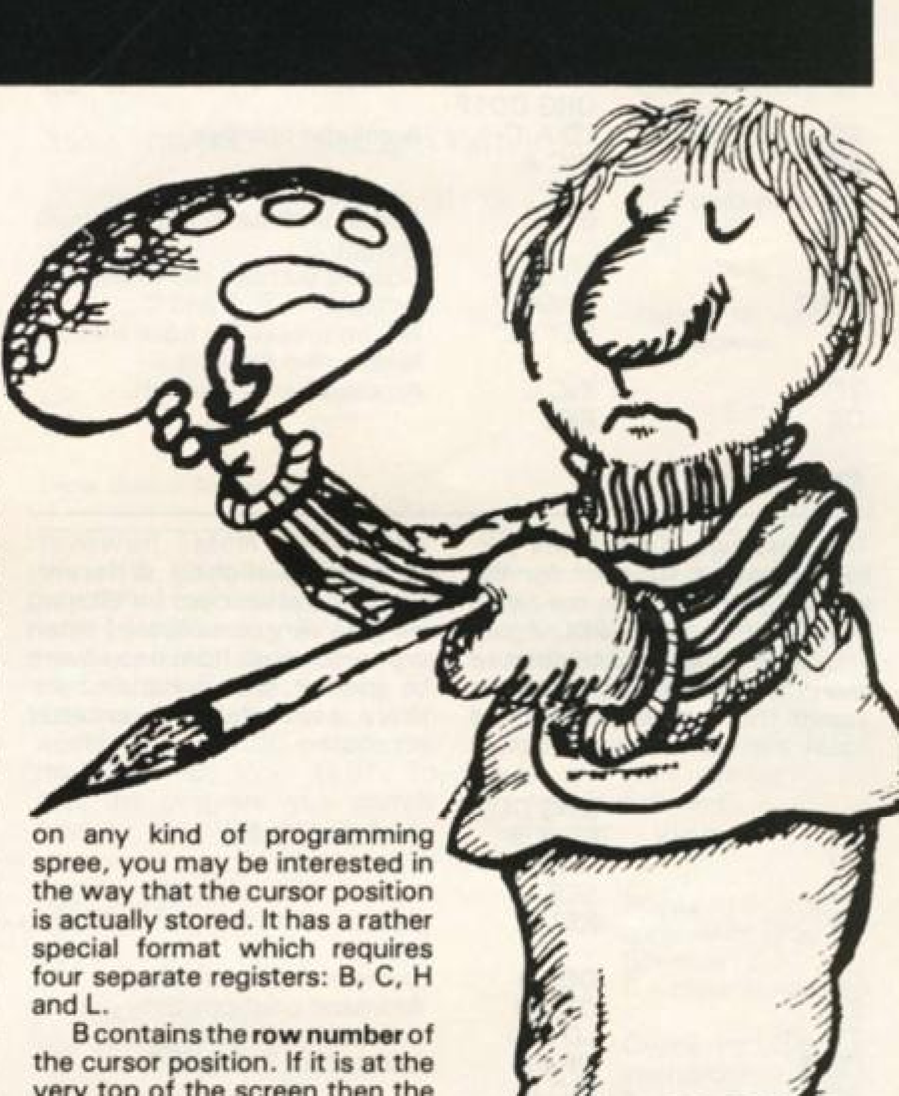
Fig. 4 Alternative lines.

```
T=NOT(Z)  =NOT(Y>=Q AND ABS(X-P)<2)
          =NOT(Y>=Q) OR NOT(ABS(X-P)<2)
          =(Y<Q) OR ABS (X-P)>=2
```

Fig. 5 Developing an alternative expression for T = NOT(Z)

Light Screen Designer Part 2

Toni Baker adds more to our great Spectrum graphics package



Hello again. This is part two of a very, very long program called Light Screen Designer. When it's all finished, what you'll end up with is a grand artwork program which will enable you to draw any kind of picture on the screen with ease. At present the program is unusable as such, and will continue to be completely unusable until you've got parts one, two and three. However, in the meantime, I'd like to build it up a little more.

Firstly, there are two mistakes in the last issue. In FIGURE ONE (keyboard overlay diagram) the key marked "USR" should be marked "TRIANGLE", the key marked "HIDE" should be marked "USR", and to the right of the key incorrectly marked "HIDF" there should be another key (yes, there are ten keys on each row) marked "HIDE". There is also a bug at the end of part one, which was entirely my fault. We require the A register to be preserved whilst the lower part of the screen is cleared, since it contains the character input by the user. The following code, overwriting the very last instruction, will cure this bug.

```
          ORG DD0D
F5        PUSH AF
CD6E02    CALL 0D6E,CLS LOWER
F1        POP AF
C9        RET
```

Sorry about that. I guess I am only human after all. In **this** article what we are going to do is come up with some machine code which draws a cursor at any pixel position on the screen. That's all — nothing too exciting. Actually you may even find this article interesting even **without** part one because it's more or less self contained.

You can have three different types of cursor in this program: an invisible one (so that you can see the whole screen without being distracted), a single dot (ie just one pixel), or a crosswires symbol — this is a cross formed by a horizontal and a vertical line, each nine pixels in total length, intersecting at the cursor position. the choice of which cursor to draw is determined by one of our very own system variables. It's called JFLAGS.

JFLAGS is a two byte program variable which lives at address DB40. (Note. **all** of the program variables lie between DB00 and DB41). Each of its sixteen bits serves a different purpose. Bit fifteen tells us whether or not the cursor is hidden (0 if it is visible, 1 if it is hidden), and bit fourteen tells us which type of cursor to use (0 for crosswires, 1 for single dot). to make life easier for us in accessing these variables, the register IX is used and is assumed to contain the value DB40. thus IX points directly to JFLAGS, and we can use (IX+0) to refer to the low part, or (IX+1) to refer to the high part.

Before we actually embark on any kind of programming spree, you may be interested in the way that the cursor position is actually stored. It has a rather special format which requires four separate registers: B, C, H and L.

B contains the **row number** of the cursor position. If it is at the very top of the screen then the row number is zero. One pixel down and the row number is one, and so on. The very bottom of the screen has pixel number BF (hex). C contains the **column number** of the cursor position. the very left hand edge of the screen has column number zero, and the very right of the screen has column number FF.

Finally, the HL register pair contains the address of the byte within the display file which actually contains the pixel at the cursor position. Any byte in the display file, of course, contains not one but eight pixels, and so HL on is own will not determine the exact cursor position (although BC will).

Left/Right Subroutines

Let's cover the first two subroutines now shall we? They are called LEFT PIX and RIGHT PIX. What they do is this: assuming that HL and BC are properly defined as described above, then they will **alter** HL and BC to point to the pixel immediately to the left or right of the one indicated. Normally the carry will be reset by this pro-

cess, but if we try to move off the left or right edge of the screen then HL and BC will remain unchanged and the carry will be set to signal the error.

How these two subroutines work is very simple, but watch what happens when we move the pixel position across from one square to another.

```
                      ORG DD13
79      LEFT PIX      LD A,C      A: = column number.
A7                    AND A
37                    SCF
C8                    RET Z       Return with carry set if at left
                                  margin.
0D                    DEC C       Decrement column number.
79                    LD A,C      A: = new column number.
F6F8                  OR F8       Isolate bits 2, 1 and 0.
3C                    INC A
C0                    RET NZ      Return unless we have moved
                                  across two squares.
2D                    DEC L       Ammend print position.
C9                    RET
```

Left Pix

```
                      ORG DD1F
79      RIGHT PIX     LD A,C      A:column number.
3C                    INC A
37                    SCF
C8                    RET Z       Return with carry set if at right
                                  margin.
4F                    LD C,A      Column number incremented.
E607                  AND 07      Isolate bits 2, 1 and 0.
C0                    RET NZ      Return unless we have moved
                                  across two squares.
2C                    INC L       Ammend print position.
C9                    RET
```

Right Pix

The next two subroutines perform a similar function for the vertical direction. they are called DOWN PIX and UP PIX. Again they alter HL and BC to point to the pixel immediately below or above the one specified, and again the carry flag will detect any error. Note, however, something slightly different. Since the procedure for altering HL gets very complicated when we try to cross from one square to another, this is handled entirely separately by another subroutine.

```
                      ORG DD29
78      UP PIX        LD A,B      B: = row number.
A7                    AND A
37                    SCF
C8                    RET Z       Return with carry set if at top
                                  margin.
05                    DEC B       Decrement row number.
25                    DEC H       Ammend print position.
78                    LD A,B      A: = new row number.
F6F8                  OR F8       Isolate bits 2, 1 and 0.
3C                    INC A
280C                  JR Z,PIX    Jump if we have moved across
                      ADDR        two squares.
C9                    RET
```

Up Pix

```
                      ORG DD36
78      DOWN PIX      LD A,B      A: = row number
3C                    INC A
FEC0                  CP B0
37                    SCF
C8                    RET Z       Return with carry set if at
                                  bottom margin.
47                    LD B,A      Row number incremented.
24                    INC H       Ammend print position.
E607                  AND 07      Isolate bits 2, 1 and 0.
C0                    RET NZ      Return unless we have moved
                                  across two squares.
```

Down Pix

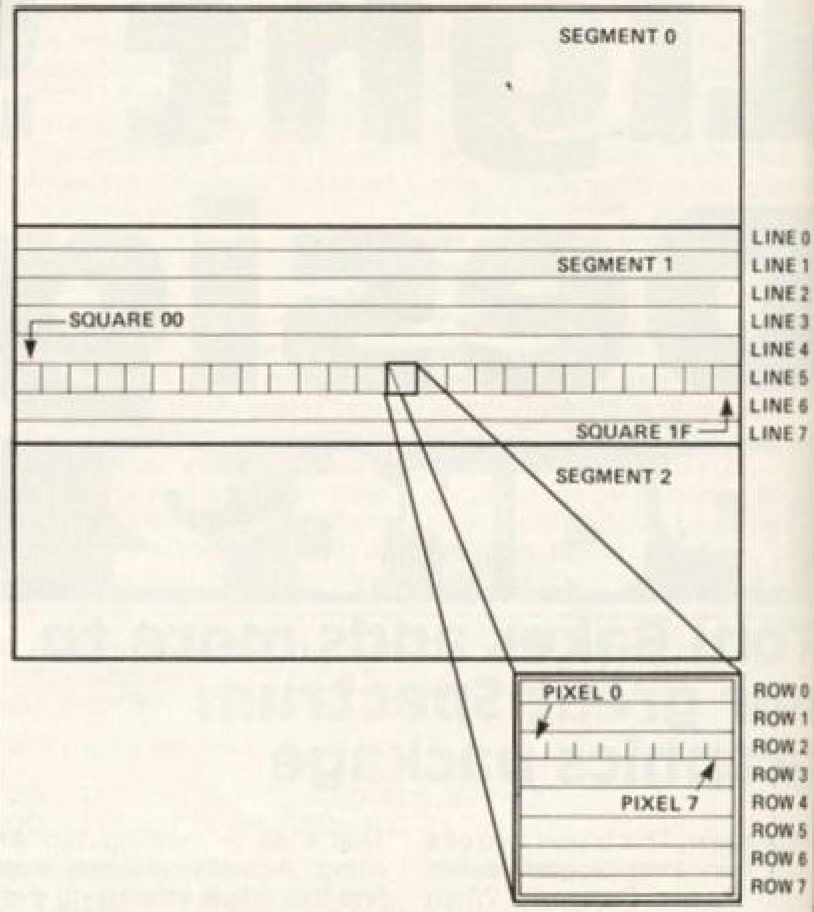


Figure 1. Spectrum display segmentation

Notice that both of the above subroutines lead to an address labelled PIX ADDR if we try to cross from one square to another. this is a subroutine which will be listed next. It actually performs more functions than are strictly necessary here, because the subroutine may be called separately from elsewhere in the program. The purpose of the subroutine is to assign HL as required, given that B and C are correct. In other words, if B contains the row number, and C contains the column number, then this subroutine will work out HL — the display file byte which contains the specified pixel — or the "print position".

In order to see how it works, some knowledge of how the screen is mapped out will be necessary.

Take a look at Figure One. It shows the screen divided into three segments, with each segment divided into eight lines, each line divided into thirty-two squares, each square divided into eight rows, and each row divided into eight pixels. If we write each such number in binary we can adopt the convention that, in general:

ss =	segment number
lll =	line number
qqqqq =	square number
rrr =	row number
ppp =	pixel number

We can work out some binary numbers straight away, for instance:

overall row number (B) =
s s lll r r r
overall column number (C) =
q q q q q p p p

What is not so obvious is the address of such a point in the display file. It is this:

print position (HL) =
0 0 s s r r r lll q q q q q

This arises because the display file is mapped out in such an unconventonal way. Nonetheless, since we now know where all the bits go in a print position address we should be able to transform values held in B and C into a value for HL. Notice that three of the bits of C (ppp) are not used in HL, since HL points to all eight pixels within the byte. The subroutine below is called PIX ADDR and performs such a transformation. Watch out for the three instruction sequence XOR B/AND mask/XOR B which creates a new byte by taking some bits from A and some bits from B according to the "mask" byte in the AND instruction.

```
                    ORG DD41
79      PIX ADDR    LD A,C      A: = column number.
07                  RLCA        Move bits 7-3 to positions
07                  RLCA        2, 1, 0, 7 and 6.
07                  RLCA
A8                  XOR B
E6C7                AND C7
A8                  XOR B       Take bits 5, 4 and 3 from B.
07                  RLCA
07                  RLCA        Move all bits into final position.
6F                  LD L,A      Assign low part of print
                                position.
78                  LD A,B      A: = row number.
0F                  RRCA
0F                  RRCA        Move bits 7 and 6 to postions.
0F                  RRCA        4 and 3.
E618                AND 18      Isolate these bits.
FE18                CP 18
C8                  RET Z       Return if both bits are set.
F640                OR 40       Assign bits 7, 5 and 5.
A8                  XOR B
E6F8                AND F8
A8                  XOR B       Take bits 2, 1 and 0 from B.
67                  LD H,A      Assign high part of print posi-
                                tion.
C9                  RET
```

Pix Addr

And now an easy subroutine. This one is designed to PLOT OVER the pixel which is addressed by B, C and HL (as before). Even this, however, has a little trick involved which you may find useful. The trick is the very short loop which manipulates the A register. Notice how is starts off with all bits reset except for one. this "set bit" is then moved by the loop until it falls in the right position. It is the XOR (HL) instruction which causes the plot to be OVER.

```
                    ORG DD5D
C5      PLOT PIX    PUSH BC
79                  LD A,C      A: = column number.
E607                AND 07      A: = pixel number within
                                square.
47                  LD B,A
04                  INC B       B: = position within row seg-
                                ment of pixel to alter.
3E01                LD A,01
0F      PPX LOOP    RRCA
10FD                DJNZ PPX    Move set bit into position.
                    LOOP
AE                  XOR (HL)
77                  LD (HL),A   Alter designated bit.
C1                  POP BC
C9                  RET
```

Plot Over Subroutine

And finally, the last subroutine for today (the one which ties all the others together). It is called DR CURSOR, which stands for DRAW CURSOR. As you can see it makes use of the bits of JFLAGS which we talked about earlier. Notice especially the code above the label DRA RET which effectively manages to CALL a subroutine at any address stored in DE without disturbing any of the registers. One point to mention is that although the subroutine relies upon IX being DB40 it does not in fact assign it.

```
                    ORG DD6C
DDCB017E            BIT 7, (JFLAGS) high
C0  DR CURSOR       RET NZ                  Return if cursor not
                                            required.
CD5DDD              CALL PLOT PIX           Plot centre of
                                            cursor.
DDCB0176            BIT 6, (JFLAGS) high
C0                  RET NZ                  Return if
                                            crosswires not
                                            required.
1113DD              LD DE,LEFT PIX
CD8BDD              CALL DR ARM             Draw left arm of
                                            crosswires.
1E1F                LD E,RIGHT PIX low
CD8BDD              CALL DR ARM             Draw right arm of
                                            crosswires.
1E29                LD E,DOWN PIX low
CD8BDD              CALL DR ARM             Draw lower arm of
                                            crosswires.
1E26                LD E,UP PIX low         Prepare to draw
                                            upper arm.
C5      DR ARM      PUSH BC
E5                  PUSH HL
3E04                LD A,04                 A:determines size
                                            of crosswires.
08      DRA LOOP    EX AF,AF'               A':stores loop
                                            count.
E5                  PUSH HL
2197DD              LD HL,DRA RET
E3                  EX (SP),HL              Push address DRA
                                            RET on stack.
D5                  PUSH DE
C9                  RET                     Call subroutine at
                                            address DE.
3807    DRA RET     JR C,DRA EXIT           Exit loop if margin
                                            reached.
CD5DDD              CALL PLOT PIX           Plot next pixel of
                                            arm.
08                  EX AF,AF'
3D                  DEC A
20EF                JR NZ,DRA LOOP          Repeat till whole
                                            arm drawn.
E1      DRA EXIT    POP HL
C1                  POP BC
C9                  RET
```

Draw Cursor Subroutine

We now have everything we need. You can test this program with the following BASIC and machine code. You can write the machine code to any address you like, and the question marks in the BASIC refer to the machine code label TEST. To test the program you merely have to input the row and column numbers of the cursor position you want to see and voila — you're away! If it works, SAVE all of this month's material (you don't need to save the test routine) together with part one (which was printed in the last issue) and wait impatiently for the next installment . . . .

Toni Baker

```
10 LET test = ?????          Points to machine
                             code label TEST.
29 INPUT b,c                 B: = row no.,
                             C = column number.
30 POKE test + 5,c
40 POKE test + 6,b           Overwrite LD BC
                             instruction.
50 RANDOMIZE USR test        Draw cursor as
                             required.
60 INPUT b,c                 Input new
                             coordinates.
70 RANDOMIZE USR test        Erase previous cursor.
80 GO TO 30                  And repeat ad
                             infinitum.
```

BASIC test Program.

```
DD2140DB  TEST   LD IX,DB40
010000           LD BC,0000         These values are
                                    overwritten by the
                                    BASIC.
CD41DD           CALL PIX ADDR      Assign HL
                                    accordingly.
CD6CDD           CALL DR CURSOR     Draw cursor at
                                    required position.
C9               RET                Return to BASIC.
```

Machine Code test programs.

MAIL ORDER PROTECTION SCHEME

If you order goods from Mail Order Advertisers in this magazine and pay by post in advance of delivery, this publication will consider you for compensation if the advertiser should become insolvent or bankrupt, provided:

1. You have not received the goods or had your money returned; and
2. You write to the publisher of this publication explaining the position not earlier than 28 days from the day you sent your order and not later than 2 months from that day.

Please do not wait until the last moment to inform us. When you write, we will tell you how to make your claim and what evidence of payment is required.

We guarantee to meet claims from readers made in accordance with the above procedure as soon as possible after the advertiser has been declared bankrupt or insolvent to a limit of £1,800 per annum for any one advertiser, so affected, and up to £5,400 p.a. in respect of all insolvent advertisers. Claims may be paid for higher amounts, or when the above procedures have not been complied with, at the discretion of this publication, but we do not guarantee to do so in view of the need to set some limit to this commitment and to learn quickly of reader's difficulties.

This guarantee covers only advance payment sent in direct response to an advertisement in this magazine (not, for example, payments made in response to catalogues, etc. received as a result of answering such advertisements:

CLASSIFIED ADVERTISEMENTS ARE EXCLUDED.

Slogo

Part II

David Nowotnik continues to run rings round us with the second part of his series for the 48K Spectrum

LOGO is a fascinating, easy to use graphics language for computers. Last time, I introduced you to a BASIC program which simulated the simplest LOGO instructions on the ZX Spectrum. These gave you a certain amount of power over an electronic turtle, which you could move around the screen, drawing lines to create simple shapes. In this, the second part of this series on LOGO, I'll be expanding the number of instructions you can give to the turtle, turning this program into a quite powerful and versatile version of LOGO.

Take a look at fig. 2. This table gives you a list of all the extra LOGO commands that will be introduced to you in this article. But before you can get the program in part one to accept any of these new commands, you'll have to add the program lines in fig. 1. to those of the original program. Here is the best way of doing that:

Additional LOGO

First, type into your Spectrum the program lines in fig. 1. Then, save this routine on tape. Load into your Spectrum the program I gave you last time, and MERGE in the new routine just saved on tape. This may seem a complicated way of adding these new lines, but it will help if you have made any mistakes in typing in the second part of the program (in third issue). If you find your expanded program does not work smoothly, then check through your second program — it should match exactly the listing in fig. 1. As I warned last issue, the merged programs will only fit a 48K Spectrum, so I'm afraid you'll only be wasting your time in the fig. 1. listing if you have a 16K machine.

When you RUN the program, you should get the same response as before; a "W:" prompt message at the base of the screen, and the turtle ("A") at the centre of the screen. If you can recall from last time, we have already used the commands FORWARD, BACK, LEFT, and RIGHT to move the turtle, and DRAW, HOME, PENUP, and PENDOWN as additional control instructions. Let's have a look at the first six of the new instructions, which also happen to be the simplest.

So far, all our drawings have been made using black 'ink' on white 'paper'. The command PENCOLOUR (or PC) allows you to change colours at anytime. To complete the PENCOLOUR command, you have to add a number, 0 to 7, according to the number keys on your Spectrum. For example, PENCOLOUR 1 will change the foreground colour to blue for subsequent movements of the turtle.

Last time, I introduced the concept of wrap. This is when an instruction to the turtle causes it to 'fall off' the edge of the screen. With wrap, the turtle will re-appear on the opposite side of the screen, and continue on its way. You can forbid the turtle to wrap round using the command NOWRAP. After using NOWRAP, if you accidentally command the turtle to go off the edge of the screen, you will get an error message. To reinstate the wrapping facility, use the command WRAP.

Turning turtle

The last three instructions of the first group are very straightforward. COPY will produce a copy of your screen display on the printer, SAVE allows you to save the program (I will deal with creating a LOGO program in part 3) on tape, and STOP simply stops the LOGO program, and hands you back to BASIC. Remember with all LOGO commands to type them out in full (do not use the equivalent BASIC keywords — LOGO won't recognise them), or use the two letter abbreviation, if one is available.

The next group of instructions all start with SET, and they move the turtle in a precise way. If you look in your Spectrum handbook (on page 121) you'll find out how high resolution graphics are created. There are 176 dots from top to bottom of the screen, and 256 from left to right. Each one of these dots can be defined by a co-ordinate (like a place can be fixed by a map coordinate). The dot at the bottom left of the screen has the coordinate 0,0 and the one at top right 255,175. If you move vertically up or down the screen, you are moving along the y axis, and moving horizontally left and right, you move along the x axis.

All that may seem horribly mathematical, but I hope it will be clearer when we use the SET instructions to move the turtle. The instruction SETX will move the turtle along the x-axis (horizontally left or right) to a specified point. Thus, SETX 10 will move the turtle from where ever it is on the screen horizontally to a point which is 11 dots (remember the first has the number 0) from the left hand edge of the screen. Similarly SETY 10 will move the turtle vertically to a point 11 dots from the bottom edge of the screen, not counting the command line. If you move the turtle with the pen down, a line will be drawn. Try this example. Reset the turtle with the DRAW instruction, then enter the command:

SX 60 SY 140 SX 128 SY 88 (and press ENTER)

This will draw horizontal and vertical lines to create a rectangle.

The command SETXY will move the turtle to the coordinates specified after the command. You'll need two numbers after SETXY, the first is the x-coordinate, the second, the y-coordinate. For example, with the rectangle from the above routine still on the screen, enter:

SETXY 60 140 (or XY 60 140) and press ENTER

This will draw a diagonal to the rectangle, taking the turtle to the coordinates 60,140 which is one corner (the top left corner) of the box.

Try a number of SETX, SETY, and SETXY instructions for yourself. The X number must be between 0 and 255, and the Y number between 0 and 175.

FULL COMMAND	TWO LETTER ABBREVIATION
PENCOLOUR	PC
WRAP	WR
NOWRAP	NW
COPY	
SAVE	
STOP	
SETX	SX
SETY	SY
SETHEADING	SH
REPEAT	RP

Fig. 2 The new LOGO commands introduced in this issue

Fig. 1 The Program Listing (SLOGO2)

```
  65 DIM x$(m,2): DIM w$(n+40,12
): DIM f$(o,12)
 110 DATA 16,22,1
 125 DATA "RP",4100,"PC",4300,"W
R",4400,"NW",4600
 130 DATA "SX",4600,"SY",4630,"X
Y",4660,"SH",4700
 135 DATA "DF",5000,"ED",5200
 215 DATA "REPEAT",4100,"]",4200
,"PENCOLOUR",4300,"WRAP",4400,"N
OWRAP",4450
 220 DATA "SAVE",4500,"COPY",452
0,"STOP",4550
 225 DATA "SETX",4600,"SETY",463
0,"SETXY",4660,"SETHEADING",4700
 230 DATA "DEFINE",5000,"EDIT",5
200
 450 DIM r(20,2)
 465 LET def=0: DIM k$(40,200):
DIM l(40)
 470 DIM b$(10,28): DIM c(10): D
IM m(40): DIM n(40)
 730 LET a$="Too many REPEATs":
RETURN
 740 LET a$="DEFINE name error":
 RETURN
 750 LET a$="No room for further
 commands": RETURN
 760 LET a$="Incorrect Command n
ame": RETURN
2140 FOR i=1 TO n+def
2210 IF i<=n THEN  GO SUB v(i):
GO TO 2220
2215 GO SUB 6000
4100 REM   REPEAT
4105 REM
4110 GO SUB 1200: IF err>0 THEN
 RETURN
4115 LET s=s+1: IF s>LEN z$ THEN
  LET err=1: RETURN
4120 IF z$(s)=" " THEN  GO TO 41
15
4125 IF z$(s)="[" THEN  GO TO 41
35
4130 LET err=1: RETURN
4135 LET rc=rc+1: LET s=s+1
4140 IF rc>20 THEN  LET err=4: R
ETURN
4145 LET r(rc,1)=s: LET r(rc,2)=
a
4150 RETURN
4155 REM
4200 REM   ] - Repeat loop
4205 REM
4210 LET r(rc,2)=r(rc,2)-1
4215 IF r(rc,2)>0 THEN  LET s=r(
rc,1): RETURN
4220 LET rc=rc-1: RETURN
4300 REM   PENCOLOUR
4305 REM
4310 GO SUB 1200: IF err>0 THEN
 RETURN
4315 IF a>7 THEN  LET err=2: RET
URN
4320 INK a: RETURN
4325 REM
4400 REM   WRAP
4405 REM
4410 LET wr=1: RETURN
4415 REM
4450 REM   NOWRAP
4455 REM
4460 LET wr=0: RETURN
4465 REM
4500 REM   SAVE
4505 INPUT "SAVE - Enter file na
me ";n$
4510 SAVE n$ LINE 2000: RETURN
4515 REM
4520 REM   COPY
4525 COPY : RETURN
4530 REM
4550 REM   STOP
4555 CLS : STOP
4600 REM   SETX
4605 GO SUB 1200: IF err>0 THEN
 RETURN
4610 IF turt=0 THEN  LET turt=1:
 GO SUB 1100
4615 IF a<0 OR a>255 THEN  LET e
rr=2: RETURN
4620 LET y2=y: LET tr=0: LET y1=
0: LET x1=a-x: LET x2=a: GO TO 3
120
4625 REM
4630 REM   SETY
4635 GO SUB 1200: IF err>0 THEN
 RETURN
4640 IF turt=0 THEN  LET turt=1:
 GO SUB 1100
4645 IF a<0 OR a>175 THEN  LET e
rr=2: RETURN
4650 LET x2=x: LET tr=0: LET x1=
0: LET y1=a-y: LET y2=a: GO TO 3
120
4655 REM
4660 REM   SETXY
4665 IF turt=0 THEN  LET turt=1:
 GO SUB 1100
4670 GO SUB 1200: IF err>0 THEN
 RETURN
4675 IF a<0 OR a>255 THEN  LET e
rr=2: RETURN
4680 LET x1=a-x: LET x2=a: GO SU
B 1200: IF err>0 THEN  RETURN
4685 IF a<0 OR a>175 THEN  LET e
rr=2: RETURN
```

```
4690 LET tr=0: LET y1=a-y: LET y
2=a: GO TO 3120
4695 REM
4700 REM   SETHEADING
4705 GO SUB 1200: IF err>0 THEN
 RETURN
4710 IF a<0 OR a>359 THEN  LET e
rr=2: RETURN
4715 LET dir=a: RETURN
5000 REM   DEFINE
5005 GO SUB 1020: IF t1=1 AND LE
N y$<2 THEN  LET err=5: RETURN
5010 IF LEN y$<>2 THEN  GO TO 50
25
5015 FOR i=1 TO m: IF y$=x$(i) T
HEN  LET err=5: RETURN
5020 NEXT i
5025 LET y$=(y$+"            ")(
 TO 12)
5030 FOR i=1 TO def+n: IF y$=w$(
i) THEN  LET err=5: RETURN
5035 NEXT i: IF def>39 THEN  LET
 err=6: RETURN
5040 CLS : PRINT "DEFINE ";y$''
5050 LET def=def+1: LET no=0
5055 INPUT "W:"; LINE z$: IF LEN
 z$>28 THEN  PRINT #1;"Too Long!
": PAUSE 200: GO TO 5055
5060 IF LEN z$<2 THEN  PRINT #1;
"Nonsense!": PAUSE 200: GO TO 50
55
5065 PRINT no;TAB 3;z$'': LET no
=no+1: LET len=LEN z$
5070 IF len<3 THEN  GO TO 5085
5075 FOR i=1 TO len-2: IF z$(i T
O i+2)="END" THEN  GO TO 5095
5080 NEXT i
5085 IF no<11 THEN  LET b$(no)=z
$: LET c(no)=len: GO TO 5055
5090 PRINT #1;"No more space is
available for  ";y$: PAUSE 200:
GO TO 5100
5095 LET len=len-3: LET b$(no)=z
$( TO len): LET c(no)=len: IF no
<10 THEN  LET c(no+1)=0
5100 LET w$(n+def)=y$: LET c$=""
: LET no=1: LET len=0
5105 IF c(no)=0 THEN  GO TO 5120
5110 LET c$=c$+b$(no, TO c(no))+
CHR$ 0: LET len=len+c(no)+1
5115 LET no=no+1: IF no<10 THEN
 GO TO 5105
5120 LET k$(def)=c$: LET l(def)=
len
5125 CLS : PRINT "STORED - ";y$
5130 PAUSE 200: CLS : RETURN
5200 REM   EDIT
5205 GO SUB 1020: IF t1=1 AND LE
N y$<2 THEN  LET err=6: RETURN
5210 LET y$=(y$+"            ")(
 TO 12)
5215 IF def=0 THEN  LET err=4: R
ETURN
5220 FOR i=n+1 TO n+def: IF y$=w
$(i) THEN  GO TO 5230
5225 NEXT i: LET err=4: RETURN
5230 GO SUB 5800
5240 GO SUB 5950
5250 PRINT #1;"1:EDIT    2:INSER
T    3:DELETE   4:REMOVE   5:RETUR
N"
5255 LET z$=INKEY$: IF z$="" THE
N  GO TO 5255
5260 IF z$<"1" OR z$>"5" THEN  G
O TO 5255
5265 LET z=VAL z$: GO TO 5200+10
0*z
5275 GO TO 5235
5300 GO SUB 5950: PRINT #1;"EDIT
 - Enter the line"
5305 LET z$=INKEY$: IF z$="" THE
N  GO TO 5305
5310 IF z$<"0" OR z$>"9" THEN  G
O TO 5305
5315 LET lin=VAL z$
5320 INPUT (lin);" W:"; LINE z$:
 IF LEN z$>28 THEN  PRINT #1;"To
o Long": PAUSE 200: GO TO 5320
5325 LET b$(lin+1)=z$: LET c(lin
+1)=LEN z$
5330 GO TO 5235
5400 IF k>9 THEN  PRINT #1;"No s
pace for another line": PAUSE 20
0: RETURN
5405 GO SUB 5950: PRINT #1;"INSE
RT - Enter line number"
5410 LET z$=INKEY$: IF z$="" THE
N  GO TO 5410
5415 IF z$<"0" AND z$>STR$ (k-1)
 THEN  GO TO 5410
5420 LET ins=VAL z$
5425 INPUT (ins);" W:"; LINE z$:
 IF LEN z$>28 THEN  PRINT #1;"To
o Long": PAUSE 200: GO TO 5425
5430 LET ins=ins+1
5435 FOR i=k TO ins STEP -1
5440 LET b$(i+1)=b$(i): LET c(i+
1)=c(i): NEXT i
5445 LET b$(ins)=z$: LET c(ins)=
LEN z$
5450 LET k=k+1: GO TO 5235
5500 LET z$="DELETE ": GO SUB 59
50: GO SUB 5900
5505 IF a$="N" THEN  GO TO 5235
5510 GO SUB 5950: PRINT #1;z$;"-
 Enter the line number"
5515 LET z$=INKEY$: IF z$="" THE
N  GO TO 5515
```

```
5520 IF z$<"Ø" OR z$>STR$ (k-1)
THEN  GO TO 5915
5525 LET k=k-1
553Ø FOR i=1+VAL z$ TO k
5535 LET b$(i)=b$(i+1): LET c(i)
=c(i+1): NEXT i
554Ø GO TO 5235
56ØØ LET z$="REMOVE ": GO SUB 59
5Ø: GO SUB 59ØØ
56Ø5 IF a$="N" THEN  GO TO 5235
561Ø LET def=def-1: IF def=Ø OR
ed=def+1 THEN  RETURN
562Ø FOR i=n+ed TO def+n
5625 LET w$(i)=w$(i+1): NEXT i
563Ø FOR i=ed TO def
5635 LET k$(i)=k$(i+1): LET l(i)
=l(i+1): NEXT i
564Ø CLS : RETURN
57ØØ LET c$="": LET no=Ø
57Ø5 FOR i=1 TO k
571Ø LET c$=c$+b$(i, TO c(i))+CH
R$ Ø: LET no=no+c(i)+1
5715 NEXT i
572Ø LET k$(ed)=c$: LET l(ed)=no
5725 CLS : PRINT "STORED - ";y$:
 PAUSE 2ØØ
573Ø CLS : RETURN
58ØØ LET ed=i-n: CLS : PRINT "ED
IT ";y$''"Please wait"
58Ø5 LET len=1: LET k=Ø: LET z$=
k$(ed, TO l(ed))
581Ø IF len>l(ed) THEN  RETURN
5815 LET c$="": LET no=Ø: LET k=
k+1
582Ø IF z$(len)<>CHR$ Ø THEN  LE
T c$=c$+z$(len): LET no=no+1: LE
T len=len+1: GO TO 582Ø
5825 LET len=len+1: LET b$(k)=c$
: LET c(k)=no: GO TO 581Ø
583Ø RETURN
59ØØ PRINT #1;z$;"- Are you sure
? (y/n)"
59Ø5 LET a$=INKEY$: IF a$="" THE
N  GO TO 59Ø5
591Ø IF NOT (a$="Y" OR a$="N") T
HEN  GO TO 59Ø5
5915 RETURN
595Ø CLS : PRINT "EDIT ";y$''
5955 FOR i=1 TO k
596Ø PRINT i-1;TAB 3;b$(i)'': NE
XT i
5965 RETURN
6ØØØ REM  Use a defined command
6ØØ5 IF count=Ø THEN  LET p$=z$:
 LET s1=s
6Ø1Ø IF count>Ø THEN  LET m(coun
t)=comm: LET n(count)=s
6Ø15 LET comm=i-n: LET z$=k$(com
m, TO l(comm))
6Ø2Ø LET count=count+1
6Ø25 LET s=Ø: GO SUB 2Ø4Ø
6Ø3Ø LET count=count-1
6Ø35 IF count>Ø THEN  LET comm=m
(count): LET z$=k$(comm, TO l(co
mm)): LET s=n(count)
6Ø4Ø IF count=Ø THEN  LET z$=p$:
 LET s=s1
6Ø45 IF count<Ø THEN  LET err=1
6Ø5Ø RETURN
```

Direction

You may have noticed that, in using these SET instructions that the turtle's direction (or heading) is not changed from that before the SET instruction was used. There is an additional instruction to alter the heading of the turtle to a definite direction; this is the SETHEADING command. Fig.3 shows you the direction the turtle will take from a given SETHEADING. SETHEADING 0 will direct the turtle to point vertically up the screen; SETHEADING 270 will point the turtle horizontally towards the left edge of the screen. The heading you give to the turtle must be between 0 and 360.

If you are confused between the SETHEADING and LEFT and RIGHT commands then remember, SETHEADING gives you *absolute* command of the heading whereas LEFT and RIGHT turn the turtle *relative* to the turtle's current direction.

One final piece of theory for this part before we try a few more examples. If you remember last time we created a square with the instructions:

FD 40 RT 90 FT 40 RT 90 FD 40 RT 90 FD 40 RT 90

You have probably noticed that the two commands FD 40 RT 90 are repeated 4 times. There is a command in LOGO which helps in repetition. Quite logically that command is REPEAT (RP for short). Here's how it is used in drawing a box. Clear the screen with the DRAW command, then enter the command:

RP 4 (FD 40 RT 90) and press ENTER

The number after REPEAT (or RP) is the number of times the commands within the square brackets are to be repeated. Remember, always leave a single space between commands, numbers, and square brackets. To be complete, the REPEAT command *must* be followed by a number, then an 'open' square bracket. A 'close' square bracket indicates the end of the repeat loop.

Nested Loops

In the same way that FOR-NEXT loops can be 'nested' in BASIC, so can REPEAT loops be nested. As an example, the following LOGO instruction will produce the symetrical pattern in fig.4.

The two loops are shown by the lines drawn above the in-

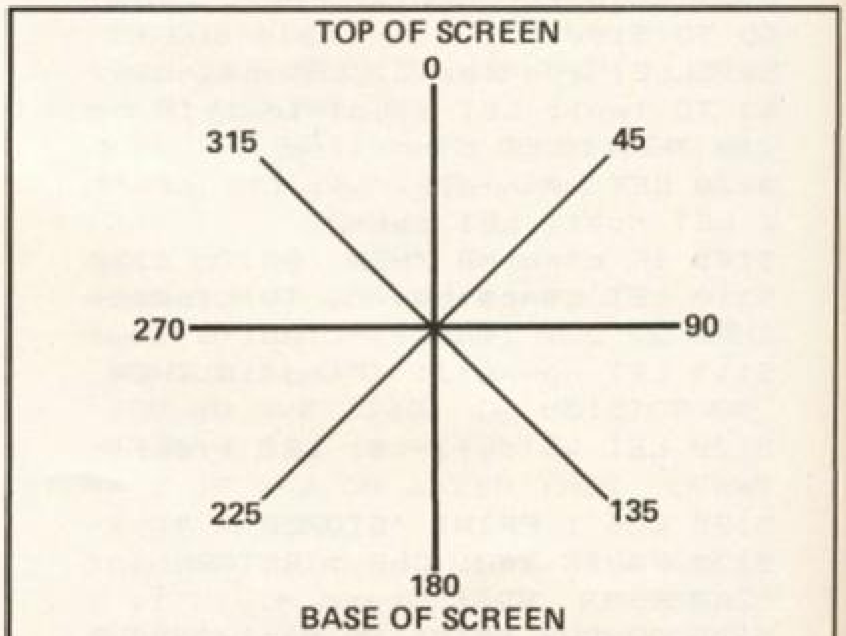


Fig. 3 Angles to be used with the SETHEADING command

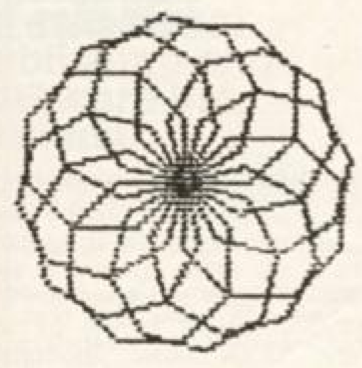
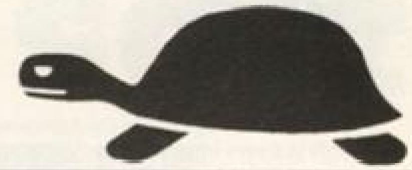


Fig. 4 An example LOGO pattern (see text)

struction. Nesting of loops is easier in LOGO than in BASIC, and as long as you have the same number of "("as")", then you shouldn't go too far wrong!

The above command can be quite simply varied to produce a variety of symmetrical shapes. If you look at the command, I've linked together (under the command) two sets of two numbers. If you multiply the first two together you should get 360, and if you multiply the second you'll also get 360. And that is the trick in getting symmetrical patterns. You can replace any pair of numbers with another pair, so long as the new numbers in the pair, when multiplied together, produce 360. Why not try a few, and see what shapes you can produce.

Here are 3 other pattern drawing routines you may like to try:



```
1.  DRAW RT 20
    RP 5 ( FD 40 RT 135 FD 40 LT 63 )

2.  DRAW PU XY 80 60 PD
    RP 8 ( FD 40 RT 45 ) RT 45
    RP 8 ( FD 96 RT 135 )

3.  DRAW PC 1 RP 3 ( FD 40 RT 120 )
    PC 2 RP 3 (BK 40 LT 60 )
    LT 90 PC 3 RP 3 ( FD 40 RT 120 )
    PC 4 RP 3 ( BK 40 LT 60 )
```

```
        ┌────────OUTER LOOP────────┐
        │                ┌─INNER LOOP─┐
RP 12 [ RT 30 RP 8 [ RT 45 FD 20 ] ]
   └─12 x 30─┘        └─8 x 45─┘
      = 360               = 360
```

Nested loops in LOGO

Type in one line at a time, pressing ENTER when you get to the end. Try and follow what is happening on the screen, and relate it to the instruction you have just entered. You'll find that you should soon understand most of the LOGO commands I've introduced so far.

Up till now, we have simply entered commands to the computer, and the Spectrum has obeyed them immediately, then the command is forgotten. It would be far more useful for the computer to store instructions and allow us to recall them at will. Our LOGO translater will then be operating on a LOGO program. The observant reader may already be aware that in this article's BASIC program, commands for program creation and modification (DEFINE and EDIT) are already present. However, space does not permit me to go into this aspect of LOGO in this issue, so you'll have to wait until the next, and final part of this series to find out how to create LOGO programs.

The ZX81 soft selection

Once again Nick Pearce voices his opinion on some of the new releases for the ZX81

Signalman International Publishing and Software Inc. (Canada).

Signalman will undoubtedly have considerable appeal to those outdoor types, scouts and boys brigaders, who are also ZX81 owners. It enables the user to learn, and practice, semaphore, blinker-lights, and Morse-code.

To start, you first select the system you wish to use from a menu. There are two Morse options, radio, wig-wag flag (in this semaphore-type system, the dots and dashes of Morse-code are represented by the posititions of the arm). There is then a further menu, from which you select up to five alternative ways to use the program.

You can run through the alphabet in your chosen system; see any letter (or number) on demand; get the computer to select letters at random; or test your knowledge by the use of a quiz. In addition, there are 50 short messages included in the program to enable the user to practice receiving signals. examples are "Welcome aboard my yacht — it is your turn to row", and "The beachmaster wants bikini suits off now". The messages can be altered by the user if necessary.

Each signalling method is displayed in the appropriate manner. In semaphore and wigwag flag, an arm-waving figure is shown in blinker-lights. The screen shows a ship with a flashing light. In Morse, in addition to an on-screen display of dots and dashes, the signals are supposed; this feature did not work at all well on my set.

Singalman is a competant and well-written program. The scout with this on his ZX81 is likely to be at a considerable advantage over his fellows, in sending and receiving coded signals. I would have liked the option to slow down the message trasnmission speed, as it is rather fast for the absolute beginner.

Speedsnake International Publishing and Software Inc. (Canada)

Another good quality, and novel, game from this Canadian company, is Speedsnake. It is a fast-moving, arcade-type game, in which the object is to steer a snake around a maze, eating nuts as you go. Nuts and berries appear at random, but the berries disappear after a short time, hence they have a higher value.

Although this may not sound too difficult, there are certain complications. The snake moves faster as the game progresses, and the length of the snake increases as the nuts are consumed. If the snake hits the wall, or another part of itself, a life is lost. If this isn't difficult enough, you can select an option to add a couple of poison balls to the maze — instant death to the snake if it eats one!

There are ten mazes in all, and you can choose to begin the game in any one of them. Once in the maze, every 32nd nut eaten by the snake puts you into the next maze-level in the game. The game also includes a personalised 'hall of fame' game scoring feature.

Speedsnake is a well-written and enjoyable game. With practice, a very high score can be achieved, and the action is fast enough to challenge even experienced arcade game Players. International Publishing and Software Inc. are at 394b, Chesswood Drive, Downsview, Ontario, Canada M3J 2W6.

Space Trek TRS Software Ltd.

Space Trek is a very good quality game in the Star Trek vein. The game is based around the ship's computer which acts as a 'menu' for the various commands which are at your disposal as Captain. Yours is the last remaining Federation starship in the galaxy, and you must destroy the Klingon fleet. You are armed with phasers and photon-torpedoes.

The galaxy is composed of 100 quadrants, and each quadrant is further divided into 100 sectors. The commands obtainable through the ships computer are: phasers, torpedoes, long and short scans, impulse engines, warp drive, status, and visual. This last mode in interactive, allowing you to move around a quadrant and fire upon the Klingon ships — assuming you have remembered to load the phasers or torpedoes! The Klingons can also fire back, of course, so beware!

Space Trek is quite a complicated game, and absorbing to play. A game can be saved at any time, and when reloaded, will continue from the point at which it was suspended.

In common with other TRS games, the cassette has been 'saved' using a fast save technique, and takes 30-40 seconds to load. A game of this length would take up to ten minutes to load at normal ZX81 loading speed. TRS claim that their fast save technique is more tolerant of volume levels than the normal ZX81 save; I certainly had no problems loading the review copy. Fast save is a tremendous advantage; waiting for some of the longer ZX81 programmes to load can be an extremely tedious affair.

Space Trek is a very good game indeed.

TRS Software are at 19, Wayside Avenue, Worthing, West Sussex, BN13 3JU.

Space Rescue D. Pinch

Space rescue is a novel arcade-type game. Your goal as pilot of a UFO, is to rescue stranded astronauts without being destroyed by meteors. You operate from a mother-ship, and have four lives per game. There are numerous large planets, some of which you can hide behind to avoid meteor storms. You gain points by shooting down meteors, and rescuing astronouts. A high-score feature keeps a record of the best score, and there are four levels of difficulty.

Space Rescue has the potential to be a very good game. However, I found it frustratingly difficult to master, and I spent quite a few trial-runs roaming the galaxy, searching for an atronaut to rescue. Clearer instructions would help in this respect. Even after learing to play Space Rescue, after considerable practice, building up a high score seemed more a matter of luck than judgement.

Still, a good idea, and quite good graphics too. I feel it could still be developed to create a very good game.

Space Rescue is available from D. Pinch, 72 Norwood Crescent, Coldbrook, Barry, South Glamorgan.

Galaxy Jailbreak Romik Software.

Galaxy Jailbreak is another good game from Romik. In this one, you control a missile base at the bottom of the screen. You can move left or right, and of course, you can fire.

The game starts with a number of evil generals in a walled area near the top of the screen — the 'jail'. Alien soldiers attempt to free the generals by hyperspacing onto the screen and hitting the jail walls, removing bricks in the process. Once on the screen, the soldiers bounce between the screen walls and the jail, removing more and more bricks as they go.

Once freed, the generals drop bombs on your missile base. If you shoot a general, he explodes — turning into three soldiers!

Points are gained for hitting soldiers and generals, and the game ends when you have lost your three lives.

The game can get very frantic at times, and in common with other Romik action games, this one offers a freeze-frame option to give the harrassed player time to compose him/herself.

A high score feature is also included, but only one level of play is available. However, the game gets increasingly difficult to play as it progresses. Comprehensive, on-screen instructions are included.

Another good-quality action game from Romik Software.

Romik Software are at 272, Argyll Avenue, Slough, Berks.

Thinchars

S J Patrick of Harrow presents this useful program that puts the *squeeze* on your screen.

THINCHARS is a machine code routine written for the Spectrum which displays 42 characters on each line of the screen, instead of the usual 32. Each character is 6 pixels wide instead of 8, including a blank column between letters. The shapes of the 96 characters (codes 32 to 127) are stored in a table at the end of the routine. Each character takes 5 bytes, one per column, which are read from left to right; the sixth column is always blank and is not stored in the table. The routine is *relocatable*, which means that it can be loaded at any address in memory and will work without modification. This makes use of the fact that when a "USR n" statement is executed in BASIC, the bc register is loaded with n, the start address of the routine called.

THINCHARS takes the text to be printed from the 1-dimensional character array s$, which must be set up by a DIM s$(...) statement at the start of the program to ensure that it is the first variable in the variables area. (An ordinary string variable would be moved to the end of the variables area each time its value was altered.) If the first variable is not array s$, the routine returns without doing any printing. The first two characters of s$ are not printed, but their codes are used as the x and y co-ordinates of where the printing is to begin; the x co-ordinate is the same as that used for PLOTting, but they value runs from 0 for the top row of the screen to 184 for the bottom row (to allow printing on all 24 rows), and is rounded off if necessary to a multiple of 8. The end of the string to be printed is signalled by a character with code 128. So a general statement to set up the string would be LET s$ = CHR$ x + CHR$ y + "text" + CHR$ 128. There is no limit to the length of s$; if the printing reaches the foot of the screen, it continues at the top.

Any of the colour items (PAPER, INK, INVERSE, OVER, etc) can be set before THINCHARS is called. Two ROM routines are used: call 3405 copies the permanent screen attributes to the temporary ones; and call 3035 uses the screen address in the hl register to set the attributes in the appropriate square. (This is done at the left hand and right-hand edges of each character, as they may be in different attribute squares.) The routine will fill the screen with 1008 characters in about 1.4 second.

Loading the routine

The routine is 199 bytes long, followed by a character table of 480 bytes; suitable start addresses at 64600 for a 48K Spectrum and 31900 for a 16K Spectrum, but any address may be used, and when the routine has been saved (SAVE "thinchars" CODE 64600,679) it may be reloaded to a different address. Remember to enter CLEAR n, where n is less than the chosen start address, before loading the routine. The following BASIC program may be used for loading: (addresses are in decimal, bytes in hex; enter x to go back one byte, s to stop)

```
10 DEF FN h (p$)=CODE p$ - 48 - 7 * (p$  "9")
   - 32 * (p$  ="a")
20 DIM a$ (2)
30 INPUT "Start address:";a
40 INPUT (a), LINE a$
50 IF a$(1)="s" THEN STOP
60 IF a$(1)="x" THEN LET a=a-1: GO TO 40
70 POKE a, 16 * FN h(a$(1))+FN h(a$(2))
80 PRINT a; TAB 6; a$
90 LET a=a+1: GO TO 40
```

Loading the character table

Figure 1 shows the 480 bytes of the character table, and the characters which they represent. They are loaded into memory immediately following the routine. Alternatively, the characters may be generated by means of the program THIN.GEN. In this, you are given a 5 × 8 grid in which to create a character, which is then stored in the appropriate place; the program displays all the characters so far loaded, and any of them may be returned to the grid for modification.

Fig. 1 Hex dump of character table.

```
ØØ ØØ ØØ ØØ ØØ ØØ ØØ 7A
ØØ ØØ ØØ 7Ø ØØ 7Ø ØØ 24
7E 24 7E 24 12 2A 7F 2A
24 32 34 Ø8 16 26 2C 52
5A 24 ØA ØØ 1Ø 6Ø ØØ ØØ
ØØ ØØ 3C 42 ØØ ØØ 42 3C
ØØ ØØ 1Ø 54 38 54 1Ø Ø8
Ø8 3E Ø8 Ø8 ØØ Ø1 Ø6 ØØ
ØØ Ø8 Ø8 Ø8 Ø8 Ø8 ØØ Ø6
Ø6 ØØ ØØ Ø2 Ø4 Ø8 1Ø 2Ø
3C 46 5A 62 3C ØØ 22 7E
Ø2 ØØ 26 4A 4A 4A 32 44
52 52 52 2C ØC 14 24 7E
Ø4 72 52 52 52 4C 3C 52
52 52 ØC 4Ø 42 44 48 7Ø
2C 52 52 52 2C 3Ø 48 48
48 3E ØØ ØØ 24 ØØ ØØ ØØ
Ø2 2C ØØ ØØ ØØ Ø8 14 22
ØØ 14 14 14 14 14 ØØ 22
14 Ø8 ØØ 3Ø 4Ø 4A 48 3Ø
3E 41 4C 52 3E 3E 48 48
48 3E 7E 52 52 52 2C 3C
42 42 42 24 7E 42 42 24
18 7E 52 52 52 42 7E 5Ø
5Ø 5Ø 4Ø 3C 42 42 4A 2C
7E 1Ø 1Ø 1Ø 7E ØØ 42 7E
42 ØØ Ø4 Ø2 Ø2 Ø2 7C 7E
1Ø 18 24 42 7E Ø2 Ø2 Ø2
Ø2 7E 2Ø 18 2Ø 7E 7E 2Ø
18 Ø4 7E 3C 42 42 42 3C
7E 48 48 48 3Ø 3C 42 4A
44 3A 7E 48 48 4C 32 22
52 52 52 4C 4Ø 4Ø 7E 4Ø
4Ø 7C Ø2 Ø2 Ø2 7C 78 Ø4
Ø2 Ø4 78 7C Ø2 1C Ø2 7C
46 28 1Ø 28 46 6Ø 1Ø ØE
1Ø 6Ø 46 4A 52 52 62 ØØ
```

```
ØØ 7E 42 ØØ 2Ø 1Ø Ø8 Ø4
Ø2 ØØ 42 7E ØØ ØØ 1Ø 2Ø
7E 2Ø 1Ø Ø1 Ø1 Ø1 Ø1 Ø1
12 3E 52 52 42 Ø4 2A 2A
2A 1E 7E 12 12 12 ØC 1C
22 22 22 12 ØC 12 12 12
7E 1C 2A 2A 2A 1A ØØ 3E
5Ø 5Ø ØØ 18 25 25 25 3E
7E 1Ø 1Ø 1Ø ØE ØØ 12 5E
Ø2 ØØ Ø2 Ø1 Ø1 5E ØØ 7E
Ø8 14 22 ØØ ØØ 7C Ø2 Ø2
ØØ 3E 2Ø 1E 2Ø 1E 3E 2Ø
2Ø 2Ø 1E 1C 22 22 22 1C
3F 24 24 24 18 18 24 24
24 1F 1E 2Ø 2Ø 2Ø 1Ø 12
2A 2A 2A 24 2Ø 7C 22 22
ØØ 3C Ø2 Ø2 Ø2 3C 38 Ø4
Ø2 Ø4 38 3C Ø2 1C Ø2 3C
22 14 Ø8 14 22 39 Ø5 Ø5
Ø5 3E 22 26 2A 32 22 ØØ
1Ø 2C 42 ØØ ØØ ØØ 7E ØØ
ØØ ØØ 42 2C 1Ø ØØ 4Ø 8Ø
4Ø 2Ø 4Ø 3E 49 55 41 3E
ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ
```

Description of program THIN.GEN

10	Ensures that s$ is the first variable
130-160	Obtains start address of THINCHARS; checks first character and halts if it is incorrect
200-260	Sets up a copy of the character set in s$; the BEEP in line 250 is to convince the user that something is happening
270-290	Print box for character, character set and instructions
300	Set cursor to top left-hand corner
310	Print cursor
320-330	Wait for a key; remove cursor
340	Plot or unplot a square in the grid
370-390	Move cursor
400-496	Load grid with character i$. Line 460 tests if the bit read from the character table is the same colour as the relevant square on the grid, and if not, changes it. POINT was used instead of SCREEN$ as the latter treats both white and black squares as spaces.
500-595	Store grid in character table in place determined by i$, and re-print the character set
600-620	Prints the character set (on yellow PAPER, so that the user can see how many empty spaces are left).

machine code

The addresses and the numbers in the assembler listing are in decimal, and the addresses are relative to the start of the program.

000	C5	push bc	save routine start address in ix
001	DDE1	pop ix	
003	FDCB0286	res 0, TV-FLAG	copy permanent attributes to temporary attributes
007	CD4D0D	call 3405	
010	012700	ld bc, 39	set ix to base address of character table
013	DD09	add ix, bc	
015	2A4B5C	ld hl,(VARS)	find first BASIC variable
018	7E	ld a,(hl)	
019	FED3	cp 211	return if not array s$
021	C0	ret nz	
022	010600	ld bc,6	jump over dimension information
025	09	add hl,bc	
026	5E	ld e,(hl)	load e and d with x and y co-ordinates of start position
027	23	inc hl	
028	56	ld d,(hl)	
029	23	inc hl	
030	7B	ld a,e	create mask byte showing which bit of display bytes is to be altered, and store in register c
031	E607	and 7	
033	47	ld b,a	
034	04	inc b	
035	3E01	ld a,1	
037	0F	rrca	
038	10FD	djnz 037	
040	4F	ld c,a	
041	CB3B	srl e	convert co-ordinates to address in display area
043	CB3B	srl e	
045	CB3B	srl e	
047	7A	ld a,d	
048	FEC0	cp 192	(return if y co-ordinate is off foot so screen)
050	D0	ret nc	
051	17	rla	
052	17	rla	
053	E6E0	and 224	
055	B3	or e	
056	5F	ld e,a	
057	7A	ld a,d	
058	1F	rra	
059	1F	rra	
060	1F	rra	
061	E618	and 24	
063	F640	or 64	
065	57	ld, d,a	
066	EB	ex de,hl	
067	1A	ld a,(de)	load character to be printed
068	13	inc de	increment pointer
069	FE80	cp 128	return if character had code of 128
071	C8	ret z	
072	D5	push de	
073	C5	push bc	calculates address in character table of first column of character to be printed (ix + 5 * ad)
074	EB	ex de,hl	
075	DDE5	push ix	
077	E1	pop hl	
078	4F	ld c,a	
079	0600	ld b,0	
081	09	add hl, bc	
082	09	add hl, bc	
083	09	add hl, bc	
084	09	add hl, bc	
085	09	add hl, bc	
086	EB	ex de, hl	
087	C1	pop bc	
088	0606	ld b, 6	load b with no. of columns
090	E5	push hl	set attributes on screen where first column of character is to be printed
091	D5	push de	
092	CDDB0B	call 3035	
095	D1	pop de	
096	E1	pop hl	
097	1A	ld a,(de)	load character pattern
098	D5	push de	
099	05	dec b	if last column then set attributes on screen
100	2006	jr nz,108	
102	E5	push hl	
103	CDDB0B	call 3035	and clear accumulator to print blank column between characters
106	E1	pop hl	
107	AF	xor a	
108	04	inc b	
109	FDCB5756	bit 2, P-FLAG	if INVERSE is set then invert character pattern
113	2801	jr z, 116	
115	2F	cpl	
116	1E08	ld e,8	number of rows
118	57	ld d,a	store pattern in d
119	FDCB5746	bit 0,P-FLAG	OVER?
123	280D	jr z, 138	
125	CB12	rl d	(OVER = 1) if pattern bit is set then flip the bit in the display
127	3003	jr nc, 132	
129	7E	ld a,(hl)	

THINCHARS

```
130 A9      xor c
131 77      ld (hl),a
132 24      inc h          move down one row
133 1D      dec e          go back for next row
134 20F5    jr nz, 125
136 1810    jr 154
138 CB12    rl d           (OVER=0) make the bit in
140 7E      ld a,(hl)        the display equal to
141 3003    jr nc, 146       the pattern bit
143 B1                     or c
144 1803    jr 149
146 2F      cpl
147 B1      or c
148 2F      cpl
149 77      ld (hl),a
150 24      inc h          move down one row
151 1D      dec e          go back for next row
152 20FO    jr nz, 138
154 CB09    rrc c          set mask and address for
156 3001    jr nc, 159       next column
158 2C      inc 1
159 7C      ld a,h         move back up 8 rows
160 D608    sub 8
162 67      ld h,a
163 D1      pop de         increment pointer to
164 13      inc de           character table
165 10BA    djnz 097       next column
167 D1      pop de
168 3E1F    ld a, 31       if x co-ordinate    248...
170 A5      and 1
171 FE1F    cp 31
173 2094    jr nz,067      (next character)
175 79      ld a,c
176 FE40    cp 64
178 308F    jr nc,067      (next character)
180 OE80    ld c, 128      ... then start a new row of
182 2C      inc 1            characters
183 208A    jr nz, 067     if a new third of the screen
185 7C      ld a,h           then update screen
186 C608    add a,8          address
188 67      ld h,a
189 E618    and 24         if off foot of screen then
191 EE18    xor 24           go to top of screen
193 2080    jr nz, 067     (next character)
195 2640    ld h,64        top of screen address
197 18EB    jr 178         (next character — jumps
                             via 178 as 067 is out
                             of range)
```

```
100 REM  THIN.GEN @ S J Patrick
110 DIM s$(99)
120 PRINT "     THIN CHARACTER G
ENERATOR"
130 INPUT "Start address of THI
NCHARS      routine?  ";base
140 IF  PEEK base=197 THEN  GO
TO     200
150 PRINT "THINCHARS not loaded
 at ";base
160 STOP
200 REM ----------------load s$
210 LET s$(1)= CHR$ 0
220 LET s$(2)= CHR$ 8
230 FOR i=32 TO 128
240   LET s$(i-29)= CHR$ i
250   BEEP .002,0
260 NEXT i
270 GO SUB 600
280 PLOT 103,136: DRAW 41,0:
   DRAW 0,-65: DRAW -41,0:
   DRAW 0,65
290 PRINT  AT 15,0;"Press 5,6,7
,8 to move cursor,        z to
plot/unplot a square,     s to
store the grid,           l to
load the grid."
300 LET x=0: LET y=0
310 PRINT  OVER 1; AT 5+y,13+x;
   "X"
320 PAUSE 0: LET i$= INKEY$
330 PRINT  OVER 1; AT 5+y,13+x;
   "X"
340 IF i$="z" THEN  PRINT  AT 5
+y  ,13+x; OVER 1;"■": GO TO
   310
350 IF i$="l" THEN  GO TO 400
360 IF i$="s" THEN  GO TO 500
370 LET x=x-(i$="5" AND x>0)
   +(i$="8" AND x<4)
380 LET y=y-(i$="7" AND y>0)
   +(i$="6" AND y<7)
390 GO TO 310
400 REM --------------load grid
410 INPUT "Character to load (E
NTER to    return):"; LINE i$
420 IF i$="" THEN  GO TO 310
430 FOR x=0 TO 4
440   LET a= PEEK (base+39+5* C
ODE   i$+x)
450   FOR y=0 TO 7
460     IF (a>127) <>  POINT (10
5+8*   x,131-8*y) THEN  PRIN
T        OVER 1; AT 5+y,13+x
;"■"
470     IF a>127 THEN  LET a=a-1
28
480     LET a=2*a
490   NEXT y
493 NEXT x
496 GO TO 300
500 REM -------------store grid
510 INPUT "Character to store (
ENTER to    return):"; LINE i$
515 IF  CODE i$<32 THEN  GO TO
   310
520 FOR x=0 TO 4
530   LET a=0: LET b=128
540   FOR y=0 TO 7
550     IF  POINT (105+8*x,131-8
*     y) THEN  LET a=a+b
560     LET b=b/2
570   NEXT y
580   POKE base+39+5* CODE i$+x
,a
590 NEXT x
595 GO SUB 600: GO TO 300
600 REM -------print characters
610 PAPER 6: LET l= USR base
620 PAPER 7: RETURN
```

Fig. 2 Alternative BASIC program to generate new character-set.

The final frontier

The ZX80 is alive and well and lives in Bath with programmer Peter Austwick. His challenging graphics game is a must!

Your brief is to take charge of the landing of the first interstellar expedition's survey probe on the planet Htrae — the home of a backward race. Being a remote controlled probe operated from a base ship many parsecs distant, there is a time lag which causes your craft to respond sluggishly. Added to this is the problem that the planet is extremely dense (like its inhabitants!) and causes the probe to accellerate rapidly. The approach velocity must be kept below 300 or you will burn up, and having mastered that little lot, you must manoeuvre the proble until you find a suitable place to land among the rocks.

Displayed on the screen is a readout of the instruments and a graphic representation of the probe and the terrain. When the cursor flashes input the main rocket thrust — this is deducted from both velocity and the fuel reserve. Next input the sideways thrust-plus or minus- to move the craft east or west. These inputs will not take effect until the next move. If the horizontal thrust is more than 9 then the terrain will change. After each set of inputs the screen will dissolve and then present an updated display. Although the descent must not exceed 300, you will have to reduce the final landing velocity to 50 or less in order to survive the landing. And, of course, the site must be flat!

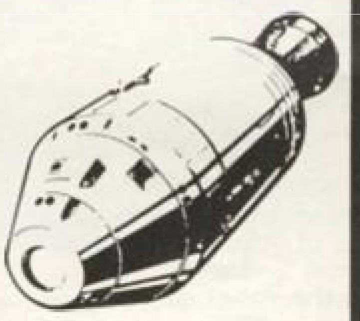
It is possible to land using only 2000 units of fuel, but it will not seem feasible at first. Once you have mastered the game, you can increase the difficulty level by introducing a random element into the gravity variable "A" or by adding the line:

450 IF H>600 THEN GOTO 110

This makes the clouds swirl as you are about to land.

The ZX80 boldly approaches the final frontier of RAM, so any additions may cause some strange screen displays — a final effort on the part of the natives to foil the invader.

(All references to any planet living or dead are completely intentional, but not to be taken too seriously!)



```
10 DIM Y(20)
20 LET U=0
30 LET M=0
40 LET T=0
50 LET L=0
60 LET H=1700
70 LET V=70
80 LET A=30
90 LET F=3000
100 LET D=5
110 FOR K=1 TO 19
120 LET Y(K)=RND(3)
130 NEXT K
140 LET Y(20)=3
200 CLS
210 LET F=F-T-ABS (L)*10
220 LET V=V+A-T
230 LET H=H-V
240 LET Z=16-(H/100)
250 LET A=A+Z*Z
260 LET D=D+L
270 IF D>20 THEN  LET D=20
280 IF H<0 THEN  LET H=0
290 IF F<0 OR V>300 THEN  GO TO 500
300 IF H=0 AND V>50 THEN  GO TO 500
310 IF H=0 AND V<51 THEN  GO TO 600
350 GO SUB 700
360 PRINT CHR$ (8);CHR$ (136)
370 FOR K=Z TO 15
380 PRINT
390 NEXT K
400 GO SUB 800
410 LET T=U
420 LET L=M
430 INPUT U
440 INPUT M
450 IF L>9 THEN  GO TO 110
460 GO TO 200
500 LET Z=15
510 GO SUB 700
520 PRINT CHR$ (136);CHR$ (8)
530 GO SUB 800
540 PRINT
550 PRINT "PROBE CRASHING"
560 STOP
600 IF Y(D+1)=3 OR Y(D+2)=3 THEN  GO TO 500
610 GO SUB 700
620 PRINT CHR$ (8);CHR$ (136)
630 GO SUB 800
640 PRINT
650 PRINT "PROBE LANDING"
660 STOP
700 PRINT "HEIGHT","SPEED","ACCLN","FUEL"
710 PRINT H,V,A,F
720 FOR K=1 TO Z
730 PRINT
740 NEXT K
750 FOR K=1 TO D
760 PRINT " ";
770 NEXT K
780 RETURN
800 FOR K=1 TO 20
810 IF Y(K)=3 THEN  GO TO 840
820 PRINT " ";
830 GO TO 850
840 PRINT CHR$ (128);
850 NEXT K
860 PRINT
870 RETURN
```

Bookshelf

Patrick Cain browses, and reports his findings.



Beginners and assembler

Two Books from McGraw Hill, one for each of the Sinclair machines.

The ZX81/TS1000 Home Computer Book by David C. Foyt is a mixture of computing made easy and a reference manual. As with so many of these publications its really a case of finding the particular author or style of presentation which appeals to you. At £9.95 its not cheap but worth looking at as it may be just what suits you. 307 pages.

Spectrum Assembly Language Programming for Arcade Games by Stuart Nicholls at £17.95 is 160 pages of advanced programming techniques.

Speed is essential for most games and many simulations and this book concentrates on getting the most out of the machine. It is intended that it will be used in conjunction with an assembler and that the reader already has some familiarity with assembly language programming.

The idea, and one I applaud in theory, is that there are plenty of books which teach Machine code mechanics but few indeed that demonstrate techniques of programming with it.

The book contains several subroutines which can be used in the readers own games and include Colour printing, Plot, Draw, Circle, Counting, Random Nos. Keyboard, Movement, Sounds, Attr. Point and Screen$, The printer, Converting programs. Plus a breakdown of routines and the usual pleathoria of tables.

A tape of the programs is available and, surprise surprise, all the programs are compatible with the McGraw Hill "ZX Spectrum Machine Code Assembler".

Just the ticket for the programmer who has progressed beyond BASIC but has not yet reached the stage where he/she automatically thinks in HEX when programming.

SH Microdrive mysteries

From Granada Publishing Ltd. we received "Make the Most of Your ZX Microdrive" by Ian Sinclair at £4.95 which is full of advice on using the latest wonder to the full.

Chapter titles include Caring for Tapes, BASIC filing Techniques, Interface Uses and a Database program. A book for those who want to investigate the possibilities of their equipment rather than just load Space Destruction in ultra quick time.

In fact there is nothing on transferring commercial programs to microdrive, and that is unfortunately probably the most usual intention of its owners (would be mine if I had one).

For the microdriver who wishes to transfer programs as well as improve his usage of the machine, I was sent three pages of typed A4 paper headed MICRODRIVE EXCHANGE.

At first glance this looked unimpressive, but on further reading this turned out to be issue 1 of a new Newsletter Club run by G. A. Bobker, proprieter of the ZX Guaranteed company.

The information contained by this newsletter was well written, accurate and useful, the way to transfer around 75% (by his estimation) of commercial software is clearly explained.

At this stage I must make it clear that I am not advocating making pirate copies for friends or profit, but if you have the means to get fast access to a program there is very little wrong with making a copy for your own use.

But back to Microdrive Exchange, at £5.00 membership per year for a monthly hint sheet it is well worth subscribing to. My best wishes for its success....

In Brief....

● **LEARNING BY NUMBERS** or rather screen photographs. Dorling Kindersley Ltd. have entered the computing arena with two large, slim, beautifully produced books for the Spectrum in a series called Screen Shot. Appropriately titled Step by step programming for the ZX Spectrum books 1 & 2 and written by Ian Graham most aspects of Spectrum BASIC programming is covered in simple colourful steps. Priced at £5.95 each they are worth looking at if you want easy to understand tuition.

● **USEFUL PROGRAMS** David Hole has written a book called Practical Uses for the Microcomputer in the Home, published by Interface.

The book contains fourteen programs of the gas/electricity bill and personal account type, so if you're fed up with people saying "OK so you've saved the world from alien taxman, but what USE is it?" buy this and show them for the price of £4.95.

● **HOBBIT HABIT** Probably the most frequently written about or discussed (and cussed) program is the Hobbit from Melbourne House.

Quick to recognise a good thing they have now published a book called "A Guide to Playing the Hobbit" by David Elkan at £3.95. Although only a slim volume it contains much valuable information and, such is the nature of the program, the solution given is only one of many and Melbourne House emphasise that it is not THE solution.

● **ADVANCED PROGRAMMING** Ian Sinclair has written a weighty tome in the form of introducing Z80 Assembly Language Programming, published by Newnes Microcomputer Books.

Great for the serious student, make up your own mind, if you feel happy with "The scheme which is followed is that of expressing the number as a modulus whose value lies between .10000000 and .11111111 and an exponent whose value is 10000000 added to the actual value of the exponent (all figures in binary)." then rush out with your £5.95 and buy it.

● **TERMINOLOGY EXPLAINED** The Colour Coded Guide to Microcomputers by Arthur Godman, published by Macdonald & Co at £10.95 in hardback or £6.95 in paperback can best be described as an encyclopaedia of computer terms.

For the fanatic its a valuable reference work and every school should have one.

Bookshelf

Patrick Cain browses, and reports his findings.



Beginners and assembler

Two Books from McGraw Hill, one for each of the Sinclair machines.

The ZX81/TS1000 Home Computer Book by David C. Foyt is a mixture of computing made easy and a reference manual. As with so many of these publications its really a case of finding the particular author or style of presentation which appeals to you. At £9.95 its not cheap but worth looking at as it may be just what suits you. 307 pages.

Spectrum Assembly Language Programming for Arcade Games by Stuart Nicholls at £17.95 is 160 pages of advanced programming techniques.

Speed is essential for most games and many simulations and this book concentrates on getting the most out of the machine. It is intended that it will be used in conjunction with an assembler and that the reader already has some familiarity with assembly language programming.

The idea, and one I applaud in theory, is that there are plenty of books which teach Machine code mechanics but few indeed that demonstrate techniques of programming with it.

The book contains several subroutines which can be used in the readers own games and include Colour printing, Plot, Draw, Circle, Counting, Random Nos. Keyboard, Movement, Sounds, Attr. Point and Screen$, The printer, Converting programs. Plus a breakdown of routines and the usual pleathoria of tables.

A tape of the programs is available and, surprise surprise, all the programs are compatible with the McGraw Hill "ZX Spectrum Machine Code Assembler".

Just the ticket for the programmer who has progressed beyond BASIC but has not yet reached the stage where he/she automatically thinks in HEX when programming.

SH Microdrive mysteries

From Granada Publishing Ltd. we received "Make the Most of Your ZX Microdrive" by Ian Sinclair at £4.95 which is full of advice on using the latest wonder to the full.

Chapter titles include Caring for Tapes, BASIC filing Techniques, Interface Uses and a Database program. A book for those who want to investigate the possibilities of their equipment rather than just load Space Destruction in ultra quick time.

In fact there is nothing on transferring commercial programs to microdrive, and that is unfortunately probably the most usual intention of its owners (would be mine if I had one).

For the microdriver who wishes to transfer programs as well as improve his usage of the machine, I was sent three pages of typed A4 paper headed MICRODRIVE EXCHANGE.

At first glance this looked unimpressive, but on further reading this turned out to be issue 1 of a new Newsletter Club run by G. A. Bobker, proprieter of the ZX Guaranteed company.

The information contained by this newsletter was well written, accurate and useful, the way to transfer around 75% (by his estimation) of commercial software is clearly explained.

At this stage I must make it clear that I am not advocating making pirate copies for friends or profit, but if you have the means to get fast access to a program there is very little wrong with making a copy for your own use.

But back to Microdrive Exchange, at £5.00 membership per year for a monthly hint sheet it is well worth subscribing to. My best wishes for its success....

In Brief....

● **LEARNING BY NUMBERS** or rather screen photographs. Dorling Kindersley Ltd. have entered the computing arena with two large, slim, beautifully produced books for the Spectrum in a series called Screen Shot. Appropriately titled Step by step programming for the ZX Spectrum books 1 & 2 and written by Ian Graham most aspects of Spectrum BASIC programming is covered in simple colourful steps. Priced at £5.95 each they are worth looking at if you want easy to understand tuition.

● **USEFUL PROGRAMS** David Hole has written a book called Practical Uses for the Microcomputer in the Home, published by Interface.

The book contains fourteen programs of the gas/electricity bill and personal account type, so if you're fed up with people saying "OK so you've saved the world from alien taxman, but what USE is it?" buy this and show them for the price of £4.95.

● **HOBBIT HABIT** Probably the most frequently written about or discussed (and cussed) program is the Hobbit from Melbourne House.

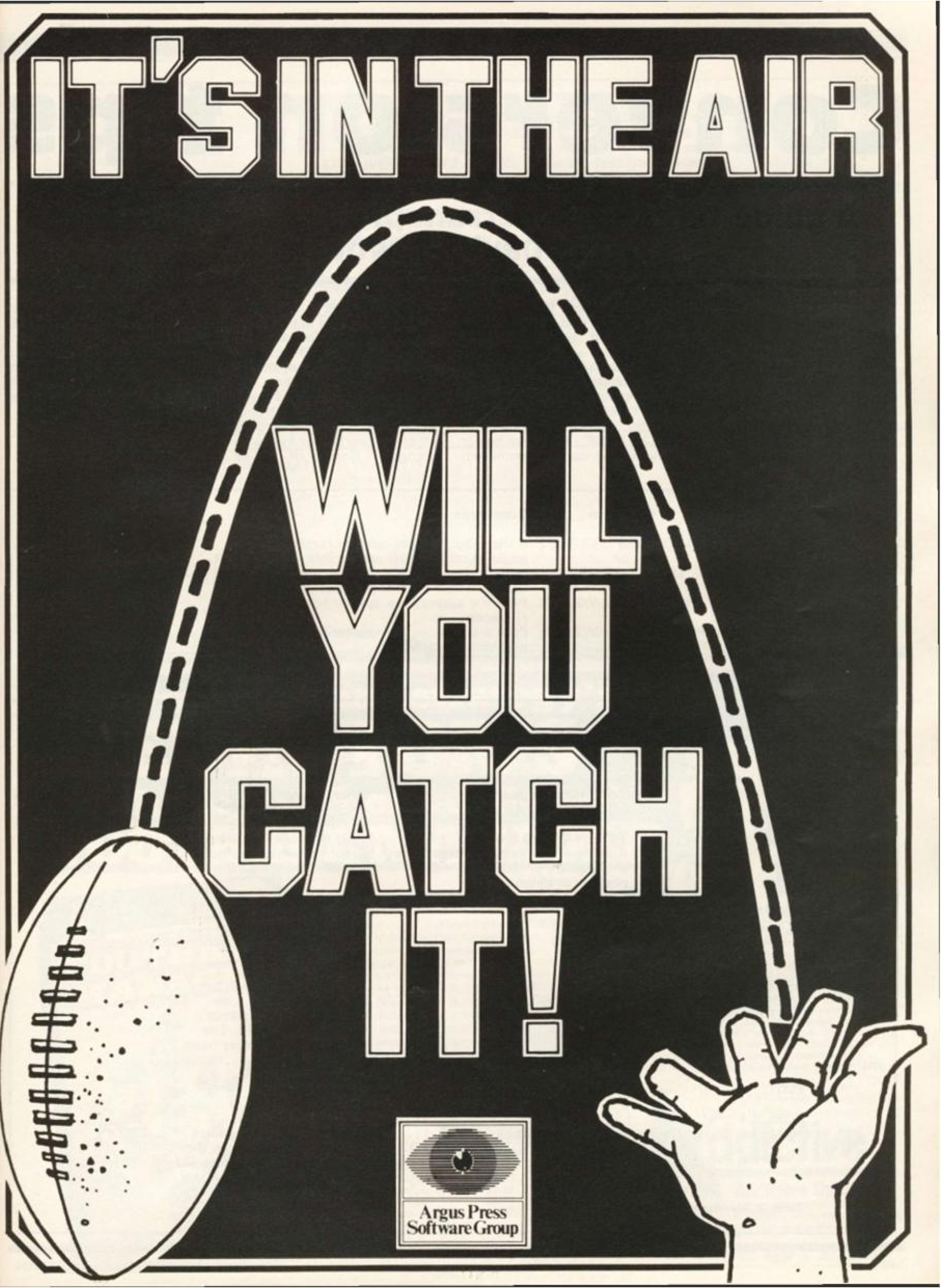
Quick to recognise a good thing they have now published a book called "A Guide to Playing the Hobbit" by David Elkan at £3.95. Although only a slim volume it contains much valuable information and, such is the nature of the program, the solution given is only one of many and Melbourne House emphasise that it is not THE solution.

● **ADVANCED PROGRAMMING** Ian Sinclair has written a weighty tome in the form of introducing Z80 Assembly Language Programming, published by Newnes Microcomputer Books.

Great for the serious student, make up your own mind, if you feel happy with "The scheme which is followed is that of expressing the number as a modulus whose value lies between .10000000 and .11111111 and an exponent whose value is 10000000 added to the actual value of the exponent (all figures in binary)." then rush out with your £5.95 and buy it.

● **TERMINOLOGY EXPLAINED** The Colour Coded Guide to Microcomputers by Arthur Godman, published by Macdonald & Co at £10.95 in hardback or £6.95 in paperback can best be described as an encyclopaedia of computer terms.

For the fanatic its a valuable reference work and every school should have one.


IT'S IN THE AIR
WILL YOU CATCH IT!
Argus Press Software Group

Conversion tips

A guide to ZX81/Spectrum program conversions from David Nowotnik.

The versions of BASIC offered by the two ZX computers are so similar that many programs for one can be used by the other. The ZX81 has only two commands which are not present on the Spectrum, SCROLL and UNPLOT, and these should cause you few problems when converting ZX81 programs to the Spectrum (see Table 1).

There are quite a lot of commands and functions on the Spectrum which are not available on the ZX81. A list of these appears in Table 4. The stars indicate those commands and functions for which there is no simple translation to ZX81 BASIC. Those for colour and sound can be ommitted; but you will have to find some alternative for the high resolution and file I/O commands.

The command PLOT appears on both computers, but the effect is quite different, so beware! Another tip: PEEK and POKE should be used with caution. In conversion, addresses will almost certainly have to be changed. Some of those changes appear in the tables. A command such as POKE USR "a"…. on the Spectrum indicates User Defined Graphics; ZX81 users don't have this facility, so you'll have to omit this and use a standard character instead.

ZX81	Spectrum	Comments
SCROLL	RANDOMISE USR 3582 or LET t=USR 3582	If the program uses random numbers, they could become rather predictable with the first option. If so, use the second, using a variable (in this case t) which is otherwise not used.
PLOT Y,X	PRINT AT 21−Y/2,X/2;	Print the appropriate quarter square graphics character.
UNPLOT Y,X	PRINT AT 21−Y/2,X/2;	Print a space, or the appropriate quarter square graphics character.

Table 1 ZX81 to Spectrum conversions.

Spectrum	ZX81	Comments
BIN eg LET y=BIN 10010101	LET y=(decimal no.) Conversion to decimal: 10010101=149 128 64 32 16 8 4 2 1 Add these numbers together when a 1 appears at the appropriate position in binary.	BIN allows the representation of a number in binary. On the ZX81 use the decimal equivalent, but beware; BIN is often used with User Defined Graphics, which are not available on the ZX81.
READ/DATA eg READ x,y DATA 50,60	LET LET X=50 LET Y=60	READ and DATA are used to store a lot of information in a program. Use LET instead.
DEF FN and FN eg DEF a(x)=SQR x LET t=FN a(i)	LET X$="SQR X" LET X=I LET T= VAL X$	The defined function can appear in a string. Use the keyword for built-in functions (eg SQR). The equivalent of FN may need 2 lines, as shown.
PLOT	no equivalent	
SCREEN$ eg LET a=SCREEN$ x,y	LET A=PEEK(PEEK 16396 +256*PEEK 16397+1+Y+33*X)	Used in interactive games to detect characters in the display file. Note — this formula only works when a RAM pack is fitted.

Table 2 Spectrum to ZX81 conversions.

ZX81	Spectrum	Comments
1 FRAMES POKE 16436,255 POKE 16437,255 LET T=(65535−PEEK 16436−256*PEEK 16437)/50	POKE 23672,0:POKE 23673,0 LET t=(PEEK 23672+256*PEEK 23673)/50 For times greater than 10 minutes, you can use byte 23674 as well.	Both computers have a counter which accurately varies by 50 every second. In the example, use the first line to start the 'clock'. The variable T will have the time in seconds after the start. The counter can only be used for 10 minutes.
2 Line number zero POKE 16510,0	POKE 23756,0 (As the start of BASIC can move, eg with microdrives) use with caution.	Converts the first line of a program to line number zero. which cannot be edited, and so is protected.
3 RAMTOP POKE 16388,X−256*INT (X/256) POKE 16389, INT (X/256)	CLEAR x	Creates a safe area at the top of RAM starting at address x, for storing data, machine code etc.

Table 3 General interconversion hints.

BEEP	*	FORMAT	*	ATTR	*
BORDER	*	INK	*	BIN	
BRIGHT	*	INVERSE		FN	
CAT	*	MERGE	*	IN	*
CIRCLE	*	MOVE	*	OVER	*
CLOSE	*	OPEN	*	POINT	*
DATA		OUT	*	SCREEN$	
DEF FN		PAPER	*	VAL$	*
DRAW	*	READ			
ERASE	*	RESTORE	*		
FLASH	*	VERIFY	*		

Table 4 Spectrum functions not available on the ZX81.

De-bugger

Getting a program typed in is often only the start of your problems. Ed to the rescue.

Typing in a program is a useful exercise. Apart from the patience required, techniques learned and the end program to be used, probably the most educational part of it is tracking down the bugs introduced by yourself or occasionally by our publication system.

In debugging you gain a much deeper insight and understanding on how the program actually works than by merely typing it in, but tracking down these errors is an art in itself and needs some skill. So here are some tips to help you in your efforts when faced with that cryptic error report!



1 NEXT without FOR
Look back through the program, either the loop has not been set up — no related FOR 'letter' = No1 TO No2 line, or the letter has been re-used as an ordinary variable within the loop with a LET 'letter' = No.

2 Variable not found
This is one of the most common errors. Again, the problem may not lie in the line where the error was detected and reported. If there is only one variable, which may be one or more letters or a string ($) variable, then that is the problem. There may be more than one variable in the line section reported and you will have to identify the offending one. In a line PRINT AT Y,X;A$ the culprit could be Y or X or A$. To find out which of them is causing the problem (it may be more than one) type in turn as a direct command:

PRINT Y Enter/Newline
PRINT X Enter/Newline
PRINT A$ Enter/Newline

Note which produces the error report. Now look back through the program printout for the line which sets it up — usually a LET or FOR command. Did you leave it out? Does the program get there or has a GOTO/GOSUB been wrongly addressed?

3 Subscript wrong
Connected with DIM A(No) or DIM A$(No). If the number in the brackets on the line where the error is reported is greater than the one in the original DIM statement, is not an integer or is less than 1, then this report is generated. If the subscript — number in brackets — is a number then check and change, however, if it is a variable then follow the procedure for tracing variables. It has probably exceeded the limits, look for lines with the variable being altered with + – * / : if necessary add limiting code. For example:

IF X >10 THEN LET X = 10

4 Out of memory
As well as for programs which are too big, it may happen if the previous program set RAMtop. Before despairing, enter CLEAR USR "a"-1 on the Spectrum: on the ZX81 SAVE the program, turn the machine off and on, then reload the program.

7 RETURN without GOSUB
Somehow the computer has reached a RETURN command other than via a GOSUB instruction. Check a GOTO hasn't been entered in place of a GOSUB. Check for a missing GOSUB.

B Integer out of range
An integer (whole number) either as a number of variable is too big or small and you are attempting to do something like PRINT AT 0,33 — not allowed! Check any variables involved as per report 2 and trace it back through the program looking for adjustments to it by + – * / : Add limiting code if needed — see report 3

E Out of DATA
A Spectrum problem. Check the number of DATA items match the number of READs; usually one (or more) has been missed out. Attempting to reread a DATA list without first using a RESTORE command will cause this and it can happen on an auto start program (saved with a LINE number). Good programming usually RESTOREs to the correct line number before using READ.

I FOR without NEXT
See report 1 but this time the NEXT is missing!

Note that the letters I have used for examples could by ANY letters not just A$,X,Y etc and depend on the particular choice of the programmer.

This is by no means a comprehensive list but I have tried to cover many of the most common error reports. Personally, I get almost as much satisfaction from debugging as I do from programming I do assure you, however, that there is absolutely no truth in the rumour that we deliberately inject bugs into our listings in order to introduce you to the dubious delights of debugging!

ZX COMPUTING

Lineage: 40p per word.

Semi display: £9.00 per single column centimetre. Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid. Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01 - 437 0699

Send your requirements to:
JANE EDMUNDS
ASP Ltd., 1 Golden Square,
London W1

SOFTWARE GAMES

SPECTRUM SOFTWARE CLUB

★★ LOWEST COST AND FREE CASSETTE ★★

- ★ Hire and exchange games from just 50p
- ★ Buy new games at the best possible discount
- ★ 100's and 100's of games to choose from
- ★ A regular newsletter with details of new releases etc.
- ★ Every new member gets a **FREE BLANK CASSETTE**

Send just £3.95 and we will rush you your membership kit and free cassette.

SPECTRUM SOFTWARE CLUB (ZX),
Video International, 287-291 High Street,
Epping, Essex. CM16 4DA. Tel: 78 78589
Or send a stamp for a catalogue.

CURRENT SPECTRUM PROGRAMS

Tradewind (£5 Cassette, £9.50 Microdrive)
Buy a ship, provisions and cargoes to trade between the harbours of the Coral Islands. Make a fortune if the weather and Pirates let you live long enough to master seamanship and economics. A strategy/adventure game with graphics for 48K.

Jersey Quest (£5 Cassette, £9.50 Microdrive)
Jersey Quest (£5 Cassette, £9.50 Microdrive)
Following an accident, you are stranded in the labyrinth of time and must find your own way back to the Present with the aid of Persons and objects from Jersey Folklore. H text-only adventure for 48K in machine code (Quill)

Morse Tutor (£4 Cassette, £8.50 Microdrive)
4-19 words/minute, variable spacing, variable number of groups of random letters, numbers or mixed random sentences, own message, single characters (including punctuation), variable Pitch, FEEDBACK via screen, Printer or SPEECH (Phonetic alphabet via Currah u-Speech on 48K version only), and repeat facility. 16K and 48K versions on one cassette, 48K only on Microdrive cartridge.
Prices exclude VAT, but include postage.

W D Software, Hilltop, St. Mary, Jersey, C.I.
Tel (0534) 81392

SPECTRUM SOFTWARE

Large range of titles credit card orders by phone or ring for further details and price list.

FABULOUS PHONES,
72 High Street, Orpington,
Kent BR6 0VQ
Tel: (0689) 74051

FROM OLD KENT ROAD TO MAYFAIR FOR 16K ZX81

Play against your computer. "This simulator has been well thought out and works well." "Good value for money" — ZX Computing Oct/Nov 83. Only **£3.00** from: **M.P.T., 42 Raedwall Drive, Bury St Edmunds, Suffolk**

"SPECTRUM SYNTHESISER" - Metronome, drum sounds, rolling chords and piano options. Play the guitar - for beginners. "Reading Music" - learning from basics with keyboard practice. All 48K Spectrum. £5.00 each. Scott Mossie, No. 11 Bungalow, Firlands, Burghfield Common, Berks.

COMPUTER SOFTWARE AND ACCESSORIES, send 2 x 16p stamps for free lists stating which computer to: Electronic Facilities 3 High Street, Sandy Beds. SG19 1AG.

SOFTWARE APPLICATIONS

SPECTRUM 48K
MOVIE-MAKING ANIMATION & MUSIC

Prog. £7 incl.
MORSE-CODE READING
Progs for ZX81 (1K) **£7** incl. and SPECTRUM 16/48K **£8** incl.
S.A.E. for full details:

Pinehurst Data Studios,
69 Pinehurst Park,
West Moors, Wimborne,
Dorset BH22 0BP

HORSE RACE FORECASTING

This super programme picked the winners of the 1983 Grand National, St Leger and many more as first choice selections.

Spectrum 48K **£12**
Spectrum 16K **£10**
ZX 81 16K **£10**
ZX 81 1K **£8**

FOOTBALL RESULTS FORECASTING

Another great programme from Butronics. It forecasts home or away wins and draws, rating each one against the others.

Spectrum 48K **£12**
ZX 81 16K **£12**

Send your cheque or PO to: **The Butronics Co., Penjerrick House, Budock, Nr. Falmouth, Cornwall TR11 5ED**

SPECTRUM KOPYKAT
IMPROVED PERFORMANCE

Simply the best. Copying **any** program is now as easy as LOADing and SAVEing your own programs. **Any ZX Spectrum** program can be backed-up onto a fresh tape. Even programs that cannot be stopped can now be copied. 100 per cent machine code. Prints file names. Copies headerless files. SPECTRUM KOPYKAT uses no program area so **full size** programs **over** 41.7K (9K for 16K machines) can be copied.

PLUS FREE Header Reader program **Despatched by first class return of post.** Send cheque/PO for **£4.95** to:

MEDSOFT, 61 Ardeen Road, Doncaster, South Yorks DN2 5ER. IT CAN EVEN COPY ITSELF.

CAR WON'T START? Won't Run Properly? Get 'Mechanic' for the 16K ZX81. Fault diagnosis program on quality cassette. Send cheques, P.O.s for £3.45 to C. Cullingworth, 16 Pamela Drive, Warmsworth, Doncaster.

SPECTRUM 48K
BOOK-KEEPING AND MONTHLY FINANCIAL ACCOUNTS
BY A. G. CLOUGH F.C.A.

A totally flexible, voucher based, COMPLETE BOOK-KEEPING SYSTEM On traditional double entry principles. BANK RECEIPTS AND PAYMENTS, PETTY CASH IMPREST, SALES JOURNAL AND LEDGER, PURCHASE JOURNAL AND LEDGER, JOURNAL ENTRIES NOMINAL LEDGE, TRIAL BALANCE.
Up to 250 Nominal Ledger accounts. Up to 400 Sales and 400 Purchase Ledger accounts. Facilitates easy preparation of MONTHLY BALANCE SHEETS and PROFIT AND LOSS ACCOUNTS. All records tangible — uses ZX Printer. Very user friendly — little expertise needed. 60 page manual. Full demonstration exercise. 9 programs-over 100K. Supplied on tape with three additional marked tapes for Data saving. Microdrive adaptable. Price £39 inc. P&P from:

MANX TAPES, GAREY VEG, GLEN AULDYN RAMSEY, ISLE OF MAN.

SICK OF "BLASTING GAMES?" BORED WITH POINTLESS "ADVENTURE GAMES?"

Then use **"Treasure Hunt Tool Kit"** to solve all anagrams, codes, etc. and win big cash prizes. Available for 48K or 16K Spectrum format. Send cheque/P.O. £4.95 to:

G. W. Components,
50 Oaktree Lane,
Mansfield,
Notts NG18 3HL.

CLOUD 10 SOFTWARE

POOLS PLAN CHECKER

For the 48K Spectrum. Fast accurate pools plan checking.
Checks Lit Plans

50 80 81 100
82 83 200 84

(16 sections) up to 240 lines £4.99 inc P&P Cheques/PO to:

CLOUD 10 SOFTWARE,
PO Box 1, Thurnby,
Leicester, LE1 9UN

SOFTWARE APPLICATIONS

Pools Prediction

Made easy on 48K Spectrum. Outputs Best Draws, Homes and Aways. Features Analytical Draw Finder which allows program to learn from it's own mistakes however small.

No fiddly database required. Prints team form comparison graphs. Analyses and graphs results every week. Takes full account of "Cup" Matches. Promotion/relegation menu.

134 Dividends won by writer so far ("This is the best and most scientific of any program I have seen" — Mr. R. A. P. Secretary, Orpington Computer Club). ("Once again thank you for one of the best investments I have made" — **Mr. D. L. B. Feltham**).

HORSE RACE
Horse Race Predictor
Very successful and easy to use predictor for Flat Racing. No knowledge of racing required. Self Teaching program is never out of date. Free permutation calculator with all programs.

Ask for:
British Pools at £8.95
Australian Pools at £8.95
Flat Racing at £11.95

Rombest, Dept. ZX.,
9 Hawkstone Avenue,
Newport, Shropshire
TF10 7SE. Tel: (0952) 813318

BACK-UP TAPE COPIER
SPECTRUM

Makes BACK-UP COPIES of *your* SPECTRUM programs easily with MANY unique features.

- M/Drive backs-up Basic/mc/ arrays on the **microdrive** STOPS programs to help the run.
- LOADS in all program parts CONTINUOUSLY — even CONTINUOUSLY — even without pressing a key. Others require many loadings — MASSIVELY saves you time and trouble and uses no program area!
- Full instructions, very user friendly. BREAK at any time then carry on LOAD. Ideal for *security* back-up copies.

Spectrum copier only £4.49 or £5.50 with M/DRIVE. (Update service for old customers £1.50 plus old tape and S.A.E.)
FULL MONEY BACK GUARANTEE

LERM (DZX), 10 Brunswick Gardens, Corby, Northants.

POOL TEAMS, results, records, analysis. One or two teams £4.95. D.A. 6 Johnson Street, Mirfield W. Yorks. S.A.E.

SPECTRUM Trade Secrets. Includes making programs, unlistable, unstoppable, invisible, how to reveal such, revealing hidden Basic in machine code programs plus hints/tips, plus microdrive supplement - £3. ZX guaranteed (office ZC), 29 Chadderton Drive, Unsworth, Bury, Lancs.

UTILITIES

STOP — REVEAL

A unique cassette for the 48K Spectrum. Genuinely lets you:

(1) **STOP** virtually all programs, **BASIC, MACHINE CODE** and headless files, protected or not.
(2) **REVEALS** hidden basic in machine code programs (could be clues in adventure games).
(3) **LISTS** actual machine code on your video.

Now you can **STOP, LIST, MODIFY, STUDY** most Spectrum programs at your leisure. (All of **STOP REVEAL** is **MICRO DRIVE COMPAT**
Includes a **DISSASSEMBLER** so can even study Sinclair **ROM** etc. plus the most advanced **HEADER READER** ever written, plus pages of **SECRETS OF THE TRADE**, explaining professional tricks to hide lines/programs etc.
You **CANNOT** get a more powerful cassette (even **STOPS/ REVEALS** programs which are genuinely **48K** long)
POST — £5.95 — FREE
(Europe plus £1.00, other areas plus £2)

KERR, 24/25 High Street, Kingslynn PE30 1BP

G. A. BOBKER Does it again

(How long before the competitors catch-up with this super program)? New TAPE 3 lets you STOP Headerless Files and unmergable Basic programs so you can study them and/or put to Microdrive. Includes a pseudo disassembler which clearly gives all the relevant info in plain English.

TAPE 3 £3.50 (£3 to club members)

ZX GUARANTEED (Dept ZX)
29 Chadderton Drive, Unsworth, Bury, Lancs. Tel: 061-766 5712

SUPERIOR TAPE COPIER

We are convinced that the new 007 SPY is the very best Spectrum tape copier available on cassette. **Find a better one offering better value for money, and TWICE your money will be refunded.**

A. Can copy multipart programs in one go automatically.
B. Can copy genuine 48K long programs.
C. Genuinly the easiest copier to use.
D. Has copied all the programs we know of.
E. Microdrive compatible AND includes program to transfer tapes to the Microdrive AND they'll work.
F. Post free even to Europe.

007 SPY IS JUST £2.95

ZX GUARANTEED (Dept ZX),
29 Chadderton Drive,
Unsworth, Bury, Lancs
Tel: 061-766 5712

SOFTWARE EDUCATIONAL

KILSOFT

EDUCATIONAL REVISION PROGRAMS
48K SPECTRUM

GCE & CSE Biology, Physics, Chemistry, Geog. and Maths. Suitable for home and school.
ONLY £4.95 EACH
For fullest details write today
Interested in ASTRONOMY?
Suitable for serious 'O' level use but also for those who have it as a hobby.
ONLY £10.00 — booklet included

32 Briarlyn Avenue, Birchencliffe, Huddersfield. Tel: (0484) 31491

LEARN ABOUT ELECTRONICS WITH YOUR SPECTRUM 48K

New structured independent learning package for beginners — designed by electronics engineers — includes:

- Computer generated animated graphics
- Learners guide
- Kit of components
- Practical guidance and exercises

Package suitable for beginners GCE/ CSE, C&G, etc and covers basic semi-conductor theory. £9.95 incl. post and packing.

E.T.S.T., 12, Kameron Avenue, Kinross.

READING AND SPELLING PROBLEMS?
THEY END HERE

For any ZX Spectrum:
5 years-Adult
WORDS and LETTERS
Features include:
★ Moving Graphics
★ Enjoyable Tunes
★ Words Changeable
★ Teacher Advised
Learning becomes fun.
2 programs, 1 on each side for only **£4.95**. Send SAE for details or make cheques/ P.O. payable to
Webco Software,
26 Nuns Orchard, Histon, Cambridge CB4 4EW.
Quick despatch. Dealer enquiries welcome.

Learning to drive?
Driver who thinks he knows it all?

LEARNER DRIVER
16K Spectrum
(FREE Highway Code)

Two fascinating educational/fun programs from a professional driving school on one tape. **Pass Your Test** — Driving Test Simulation. Are you good enough to pass first time? *AND* **Know Your Signs** — Test your knowledge of the Highway Code. Both 16K programs on tape. Unique and excellent value at an inclusive price of **£6.00**. Cheques to: **M. J. Hawkins, D.O.T.A.D.I., M.I.A.M., Stowe IX Churches School of Motoring (Est. 1974), Dept ZX, Benu Vista, Church Stowe, Northampton.**

Ed.SOFT Educational Software

76 Woodville Road, Exmouth EX8 1SW
Tel: (0395 275741)
Cassette based Educational Software for Spectrum micro. Suitable for all from 7 to 13. Programs from **£4.95** include: **Whatsit** (20 questions game), **Wordflash** (to aid the increase of reading age) **Mmaths/Placeval** (add, subtract, divide, multiply, check place values). **C. Rules** (£5.95 information, testing and reinforcement of rules for hard & soft 'c'). Also **Anagram** and ***Basically Basic*** *Booklet (£1) & others.*

POPPY PROGRAMS ZX Spectrum (16 or 48K)



HIDDEN LETTERS	Age 5-adult	**£6.00**
Highly educational reading game, 9 skill levels.		
MONEY	Age 3-10	**£6.50**
Coin recognition, 4 easy levels teach money sums		
TELLING THE TIME	Age 3-10	**£7.00**
6 programs for learning in easy stages, 12, 24 hour, digital time, time in words, clocks to set.		
LEARNING TO READ 1	Age 2-6	**£6.50**
4 All graphics Pre-Reading games		

All cassettes include special **keyboard overlays.** P&P inc.
Richmond House, Ingleton, via Carnforth, Lancs LA6 3AN

ZX81 (16K) HIDDEN LETTERS

Word game helps reading ability
£4.95 inc from
Poppy Programs

CLUBS & USER GROUPS

MICRODRIVE USERS CLUB

Called **Microdrive Exchange** this newsletter club is designed to help by the exchange of info and ideas. Did you know you can pay up to £5.50 for a cassette to transfer tapes to microdrives, and then they don't run. Issue 1 will include how to avoid this plus transferring **Vu-File ZZOOM** etc. Membership is £5 per year. Europe £8 per year.
Please make cheques payable to ZX Guaranteed. **Microdrive Exchange** is run by C. A. Bobker.

ZX GUARANTEED (Dept ZC)
29 Chadderton Drive,
Unsworth, Bury,
Lancs.

COURSES

NEW TO COMPUTING

We are here to help you gain skills in computing. Individual machines, qualified tuition, practical course. Ring **Computer Workshop on 01-318 5488. 4 Lee High Road, Lewisham, London SE13 5LQ**
Childrens' Holiday Courses also available.

FOR SALE

Machine Code Graphics in 3-D for ZX 81 16K. Co-ordinates XYZ any view angle and distance in seconds. It is a low cost C.A.D. casette and instructions only £5.95 + £1.05 Post and Packing.
Send cheque to:
Danny Hladky,
Software Research
Krimmerstr. 10
D-6050 Offenbach/M
West Germany.
Trade enquiries welcome.



Replica, loud firing Colt .45 Automatic, as used by U.S. Army, ideal stage prop with ammo £5.25,- carriage 50p.



Snub nose Police Special with silencer and ammo £4.70. Carriage 50p.



Colt .45 as used by U.S. Police Department. Ideal stage prop with ammo. £3.50, carriage 40p.

Ideal for video film making. Mail order only. Send cheques or PO to:
Razzamattazz, 80 Selhurst New Rd., London SE25

RTTY: Receive and Transmit program for Spectrum 48K. Write for info to: Lacroix, 8 Perenlaan 2400, Mol, Belgium.

JOKES



FREE

Britain's No. 1 Joke Catalogue, packed with over 500 practical jokes from 5p

Stink bombs. Whoopee cushion, wobbly lager glass, laxative tea bags, smoke bombs, willie sugar, cigarette bangers, joke blood, sick, turds, soap sweets, wet jokes, exploding jokes, magic tricks, party fun kits, masks, make-up, sea monkeys, girl strip pens, adult party packs, saucy novelties, naughty presents, posters, badges, the complete Joke Shop by Post.
Send 12½p stamp with your name and address for bumper colour catalogue and Free Gift to:
MATCHRITE, THE FUNNY BUSINESS (DEPT ZX), 167 WINCHESTER ROAD, BRISTOL BS4 3NJ

QL UNUSED £25 premium for immediate delivery 043787-644. Also Spectrum 48K hardly used Cambridge Joystick, ZX Printer with good software. Some splitting £195.

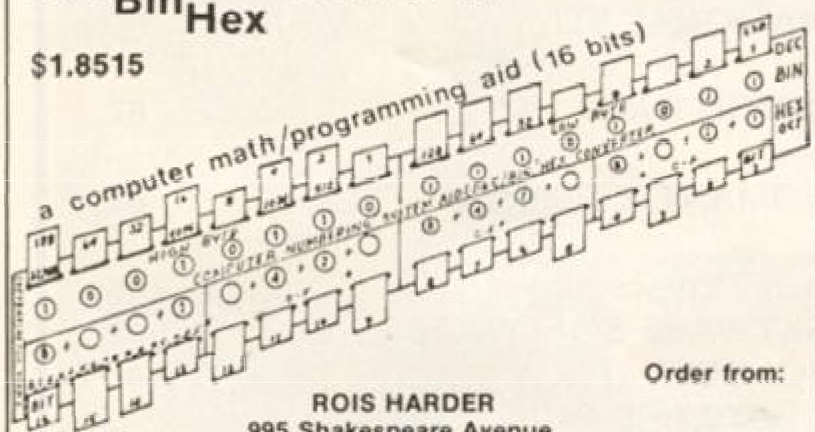
SOFTWARE

ZX SPECTRUM OWNERS. We specialise in practical programs for household business, utility and education. Send SAE/IRC for our mail order catalogue. SD Micro-Systems (ZX) P.O. Box 24, Hitchin, Herts.

SOFTWARE APPLICATIONS

Dec$_{\text{Bin}_{\text{Hex}}}$ Converter

$1.8515



Order from:
ROIS HARDER
995 Shakespeare Avenue,
N. Van., B.C., Canada V7K 1E7 (604) 980 4167

ALARMS

BURGLAR ALARM Equipment. Please visit our 2,000 sq. ft. showrooms or write or phone for your free catalogue. CWAS Ltd., 100 Rooley Avenue, Bradford BD6 1DB. Telephone: (0274) 731532.

REPAIRS

ZX81 SPECTRUM. We can now offer out-of-guarantee repairs by our Computer Dept. Our engineers have had over three years experience servicing Sinclair Computer products. Price including p&p. ZX81 £11.50, 16K Ram £9.95, Spectrum £18.75. Send with cheque or PO. T.V. Service of Cambridge, French's Road, Cambridge CB4 3NP. Phone (0223) 311371.

SPECTRUM REPAIRS
£16.95 inclusive of labour, parts, p&p etc. ZX81 for £11. Fast (av. repair 24 hrs) reliable service by qualified engineers. Send computer only to:
H S Computer Services
2 The Orchard, Warton,
Preston, Lancs PR4 1BE.
Tel: 0772 632686 or 634897

SERVICES

ZX81 & SPECTRUM REPAIRS
ADD ON SPECIALIST
MOST COMPUTER COMPONENTS
SOME SPECTRUM REPAIRS £15
- 24 hour turnaround or While-You-Wait
- **Keyboard with spacebar — £46.00**
- Open 9am to 5pm Mon-Sat

SEND FOR FREE ESTIMATE
(ENC. £1.60 P&P)
MANCOMP LTD.
Printworks Lane, Levenshulme
Manchester M19 3JP.
Tel: 061-224 1888

WANTED

SECOND HAND SPECTRUM GAMES wanted – originals only (with inserts, etc.) Disliked/Overzapped/dissappointing games – we'll buy them! Send list for offer by return! G. Inglis, 37 Cockburn Street, Edinburgh EH1 1BP.

LIBRARIES

SPECTRUM OWNERS
Why not use our software exchange service. With an extensive range of games, educational programs, etc. Only 50p per tape plus P&P.
Send SAE for full details to:
X. Lent Software,
8 Cheviots, Highfield,
Hemel Hempstead HP2 5RD

ACCESSORIES

SPECTRUM ZX81
OVERHEATING? NOISY?
KEYBOARD RUNS HOT?
Problems solved with PR adaptor, no soldering, no dismantling, just plugs in place. Please state model, **£8.95** plus 75p p&p. PR adaptor plus reset switch (allows system reset) **£10.95** plus 75p p&p. ZX supply repair service £5.95 inclusive. Cheques & PO's to: **PR Electronics, 14 Bretby Close, Doncaster DN4 6EL.** 24 hour answerphone 0302 530545/0302 539747. German stockists: **ERC Services, Heinrichstrasse 93, 4000 Dusseldorf, W. Germany**

adapt electronics

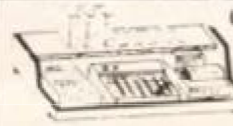
The answer to "Wobbly Rampack"

- RAMLOK KIT — *Suitable for ZX80 and 81* — £7·50
- SPACER KIT — *to use RAMLOK with ZX Printer* — £1·50
- EDGE CONNECTOR (female/Spectrum/56way ZX/48w) — £2·50
- CASSETTE LEADS — *Standard Sinclair type (pair)* — £1·50
- 8K BASIC ROM for ZX81 or update your old ZX80 — £2·00
- SINCLAIR ZX81 Programming Instruction Manual — £2·80
- 16K RAMPACK for ZX81 — made by Sinclair — £20·00
- VIDEO OUTPUT ADAPTOR for any ZX or SPECTRUM provides composite video thru' a phono socket. Full instructions NO SOLDERING NECESSARY — £2·75
- SPECTRUM RS232 Lead — for Interface I — £10·80

Please add 50p P and P per order or send SAE for further details. 01504 2840

20 STARLING CLOSE, BUCKHURST HILL, ESSEX IG9 5TN

I/O FOR YOUR SPECTRUM. Four port 64 bit for £49.95. Please send S.A.E. for details to Future Electronic Services, 55 Eastcote Avenue, South Harrow, Middx.



UNIT ONLY

Tidy units made in 3 styles and colour to suit all computers and their hardware. At £23.99 inc p&P and VAT
Send for details to:
E. R. CAMERON & Son Ltd, (Established: 1870)
H/O 12 The Vineries, Endfield, Middx, EN1 3DQ

WANT TO BEAT THE REST TO THE POST? . . . .



. . . USE ZX CLASSIFIED!! 'PHONE JANE ON 01-437 0699

ZX COMPUTING
CLASSIFIED ADVERTISEMENT — ORDER FORM

If you have something to sell now's your chance! Don't turn the page — turn to us!
Rates of charge: 40p per word per issue (minimum of 15 words).Please state classification
and post to A & B COMPUTING, CLASSIFIED DEPT, 1 GOLDEN SQUARE, LONDON W.1.

					£5.25
					£7.00
					£8.75
					£10.50
					£12.25
					£14.00

Please place my advert in **ZX COMPUTING** for ...... issues commencing as soon as possible.

I am enclosing my Cheque/Postal Order/International Money Order for: (delete as necessary) £..... (Made payable to A.S.P. Ltd)

VISA OR Debit my Access/Barclaycard (Delete as necessary) We welcome Access

All classified advertisements must be paid for in advance.

Please use BLOCK CAPITALS and include post codes.

Name (Mr/Mrs/Miss/Ms) ....
(delete accordingly)

Address ....

....

....

Signature .... **Date** ....

Daytime Tel. No. ....


ASP SOFTWARE
Presents
The Greatest Challenge Ever To Face Man And Machine
(Atari, BBC B, Spectrum and Commodore 64)
FIVE MILLION BARBARIANS
AN ARMY OR TWO OF PERSIANS
SEVERAL THOUSAND UPSET ARMENIANS
FIFTY LEGIONS OF ITINERANT ITALIANS
..................AND YOU!
THE FALL OF ROME
SPQR
In case of difficulty, Order from: Argus Press Software Group, No 1 Golden Square London W1R 3AB
Please add 50p p&p and allow 14 days for delivery
At your dealers now!
£6.99 all inc.
HORNSBY 84